
Power Electronics Questions

Basic Circuits

1. What is Ohm's Law?

Ohm's Law states that the current II flowing through a conductor between two points is directly proportional to the voltage VV across the two points and inversely proportional to the resistance RR. It is given by the equation:
V=I×RV=I×R
Do people get this wrong?

It's a basic concept, but under stress or in tricky situations, people can make mistakes or misunderstand the conditions under which it applies.

2. Conceptually Explain Ohm's Law.

Conceptually, Ohm's Law is about how voltage (the "push") across a component relates to the current (the "flow") through it and the resistance (the "opposition" to that flow). If you increase the voltage, more current flows, but if resistance is high, less current flows.

3. What are some basic circuit analysis laws?

Ohm's Law: V=I×R
Kirchhoff's Current Law (KCL): The total current entering a junction equals the total current leaving it.
Kirchhoff's Voltage Law (KVL): The total voltage around any closed loop in a circuit must sum to zero.
Power Law: P=V×I (Power is voltage times current).

4. What are KCL and KVL?

KCL (Kirchhoff's Current Law): At any junction (or node) in a circuit, the sum of currents flowing into the node must equal the sum of currents flowing out. This law is based on the principle of conservation of charge.

KVL (Kirchhoff's Voltage Law): For any closed loop in a circuit, the sum of all the voltages around the loop is zero. This law is based on the conservation of energy.

5. Basic KCL/KVL Circuit Problems

KCL Example: At a node where three wires meet, if 2A of current flows in and 1A flows out through another branch, then the current through the third branch must be 1A (to satisfy KCL).

KVL Example: In a loop with a battery of 10V and two resistors with voltage drops of 6V and 4V, the sum of voltage drops (6V + 4V = 10V) equals the battery voltage.

6. What is the equation to find power?

The power equation is: P=V×I Where:

PP is power (in watts),
VV is voltage (in volts),
II is current (in amps).

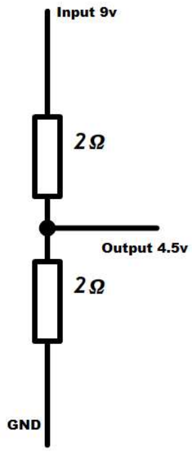
Another form using Ohm's Law: P=I2×Ror P=V2R

7. Why is power loss often due to current, not voltage?

Power loss, particularly in resistive components like wires, is proportional to the square of the current:
Ploss=I2×R This means a small increase in current leads to a large increase in power loss (due to heat dissipation).
Voltage by itself doesn't cause significant power loss unless there's current flowing through a resistive element.

8. Draw a Voltage Divider Circuit. What is the Voltage Divider Equation? Derive it.

A voltage divider consists of two resistors in series across a voltage source. The output voltage is taken across one of the resistors.
Circuit:



Voltage Divider Equation:

The output voltage Vout is given by: Vout=Vin×R2/(R1+R2)

9. Basic RLC Circuit Analysis

An RLC circuit consists of a Resistor (R), Inductor (L), and Capacitor (C).

Resonance: At a certain frequency (resonant frequency), the inductive reactance and capacitive reactance cancel each other out, leaving only the resistance.

The resonant frequency f0 is given by: f0=1/2πL×C

The circuit behaves as a low-pass filter below resonance, and as a high-pass filter above resonance.

10. Basic LPF/BPF/HPF Filter Analysis

Low Pass Filter (LPF): Passes low frequencies and attenuates high frequencies. Composed of a resistor and capacitor in series, with the output taken across the capacitor.

High Pass Filter (HPF): Passes high frequencies and attenuates low frequencies. Composed of a resistor and capacitor in series, with the output taken across the resistor.

Band Pass Filter (BPF): Allows a certain range of frequencies to pass and attenuates frequencies outside this range. Made using a combination of inductors and capacitors.

The cutoff frequency for an LPF/HPF is: fc=1/2πRC

11. Passive Components' Parasitics

Passive components like resistors, capacitors, and inductors are not ideal in the real world and have parasitic elements:

Resistors: Have parasitic inductance and capacitance.
Capacitors: Have parasitic resistance (ESR - Equivalent Series Resistance) and inductance.
Inductors: Have parasitic resistance and capacitance.

These parasitics can affect circuit performance at high frequencies.

12. Basic Op-Amp Circuit Analysis

An Operational Amplifier (Op-Amp) is a high-gain voltage amplifier with differential inputs (inverting and non-inverting).

Inverting Amplifier: The input signal is applied to the inverting input through a resistor. The output is 180° out of phase with the input. $V_{out}=-(R_f/R_{in})*V_{in}$

Non-Inverting Amplifier: The input is applied to the non-inverting input. The output is in phase with the input.
$V_{out}=(1+R_f/R_{in})V_{in}$
These amplifiers are used in various analog circuits like filters, buffers, and signal conditioning

Passive Components

Resistors

13. Draw the Symbol for a Resistor

The symbol for a resistor is a zigzag line. It can vary slightly depending on the region:

American Standard: Zigzag line.
International (IEC): A rectangle.

14. What is Resistance?

Resistance is a measure of how much a material opposes the flow of electric current through it. It's measured in ohms (Ω). The higher the resistance, the less current flows for a given voltage.

15. What Does Resistance Depend On?

Resistance depends on:

Material: Conductors like copper have low resistance, while insulators like rubber have high resistance.
Length: The longer the material, the higher the resistance.
Cross-sectional Area: The larger the area, the lower the resistance.
Temperature: For most conductors, resistance increases with temperature.

The formula for resistance is given by: $R=\rho\times(L/A)$ Where:

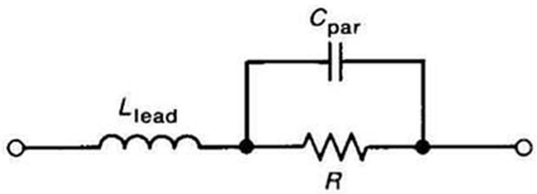
ρ is the resistivity of the material,
L is the length,
A is the cross-sectional area.

16. Draw a Realistic Circuit Model for a Resistor. What Are the Parasitics and Where Do They Come From?

In a real-world resistor, parasitics include:

Parasitic Inductance: Caused by the physical wire and layout of the resistor.
Parasitic Capacitance: Due to the proximity of the resistor's leads or materials.

Realistic model:



Where:

R is the ideal resistance.
L_{LEAD} is parasitic inductance.
C_{PAR} is parasitic capacitance.

17. What Are Some Common Resistor Values?

Resistor values follow the E-series of standard resistor values, which include:

E6 Series: 10, 15, 22, 33, 47, 68
E12 Series: 10, 12, 15, 18, 22, 27, 33, 39, 47, 56, 68, 82
E24 Series: Adds more precise values like 12.1, 18.2, 47.5, etc.

Common values are: 10Ω, 100Ω, 1kΩ, 10kΩ, 100kΩ, etc.

18. What Are Some Common Resistor Packages and Sizes?

Common through-hole and surface-mount resistor packages:

Through-hole:
Axial Lead Resistors (e.g., 1/4W, 1/2W).
Surface-Mount Device (SMD):
SMD sizes: 0603, 0805, 1206, 1210.

Axial resistor example:

bash

```
--/\/\/\/--
```

SMD resistor example:

A rectangular chip with size codes like 0805 (0.08" x 0.05").

19. What Are Some Common Failure Modes of a Resistor?

Common resistor failure modes:

Overheating: Excessive current can cause the resistor to overheat, leading to burning or complete failure.
Open Circuit: The resistor can physically break, causing it to no longer conduct.
Drift in Resistance: Over time, the resistance value may drift due to aging or environmental factors (e.g., humidity, temperature).
Short Circuit: Rarely, a resistor can fail by shorting, resulting in zero resistance.

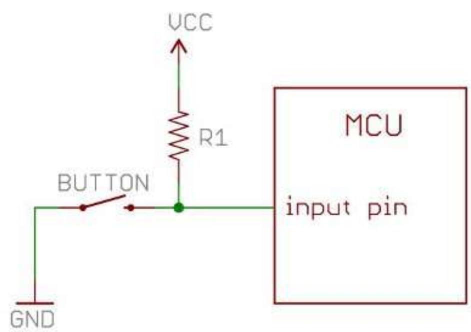
20. What Are Pullup/Pulldown Resistors? How Do You Spec Them?

Pull-up Resistors: Used to connect an input pin to a high (logic 1) voltage level when the input is not actively being driven.
Pull-down Resistors: Used to connect an input pin to ground (logic 0) when the input is not actively driven.

How to spec them:

Value Calculation: Typically, values range from 1kΩ to 100kΩ, depending on the input impedance and speed. A common value is 10kΩ.
Choosing a Value:
Higher values (like 100kΩ) reduce power consumption but may slow down the response time.
Lower values (like 1kΩ) provide faster transitions but consume more power.
Digital Logic: In digital circuits (e.g., microcontroller inputs), the pull-up/pull-down resistor value is chosen based on the input capacitance and required switching speed.

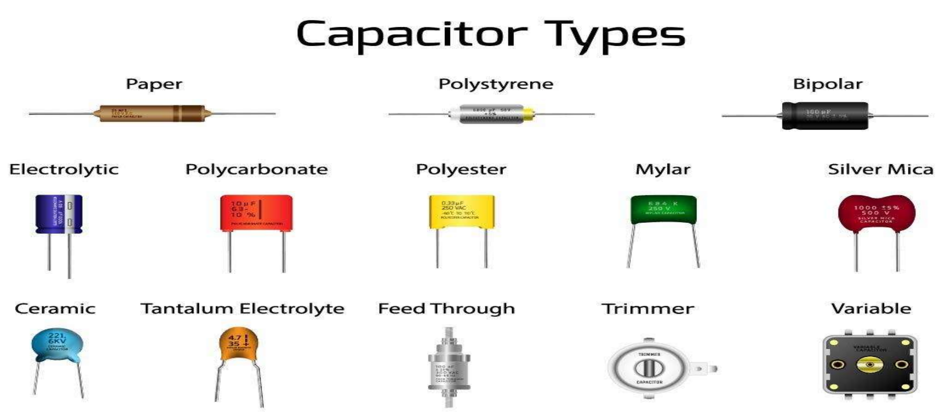
Example for a pull-up resistor:



Capacitors

21. Draw the Symbol for a Capacitor

There are two common symbols for a capacitor:



22. What is Capacitance?

Capacitance is the ability of a capacitor to store electrical charge. It's measured in farads (F). A capacitor stores energy in the form of an electric field when a voltage is applied across its plates.

23. What is a Dielectric?

A dielectric is the insulating material placed between the plates of a capacitor. It increases the capacitor's ability to store charge by reducing the electric field strength for a given charge.

Common dielectric materials include air, ceramic, mica, and plastic.

24. What is the Equation for Impedance of a Capacitor?

The impedance ZZ of a capacitor in the frequency domain is given by: $Z_C=1/j\omega C$ Where:

J is the imaginary unit,

ω=2πf is the angular frequency (with f being the frequency),
C is the capacitance.

25. What is the Differential Equation for a Capacitor?

The current II through a capacitor is related to the rate of change of voltage VV across it: I(t)=C(dV(t)/dt)This means the current through a capacitor depends on how quickly the voltage across it changes.

26. Does a Capacitor Have Positive or Negative Reactance?

A capacitor has negative reactance. Capacitive reactance XC is: XC=−1ωC This means that as frequency increases, the reactance of the capacitor decreases.

27. What is the Continuity Condition? What Do Inductors Resist Change to?

The continuity condition for a capacitor is that the voltage across it cannot change instantaneously; it takes time for a capacitor to charge or discharge.

Inductors resist changes in current. An inductor will oppose sudden changes in current through it due to its inductive property.

28. How Does a Capacitor Behave When Initially Excited and at DC Steady-State?

Initially excited: When a capacitor is first connected to a voltage source, it acts like a short circuit, allowing current to flow as it charges.
DC steady-state: At DC (zero frequency), the capacitor behaves like an open circuit because once it's fully charged, no more current flows through it.

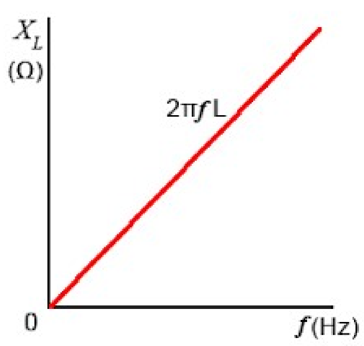
29. What is an Inductor's Impedance at DC vs Infinitely High Frequency? Plot this Behavior.

At DC: The impedance of an inductor is zero (ZL=0), acting like a short circuit.
At infinitely high frequency: The impedance becomes infinite (ZL→∞), acting like an open circuit.

The impedance ZL of an inductor is given by: ZL=jωL

Behavior: At low frequencies, the impedance is small, and at high frequencies, the impedance increases linearly with frequency.

Plot:

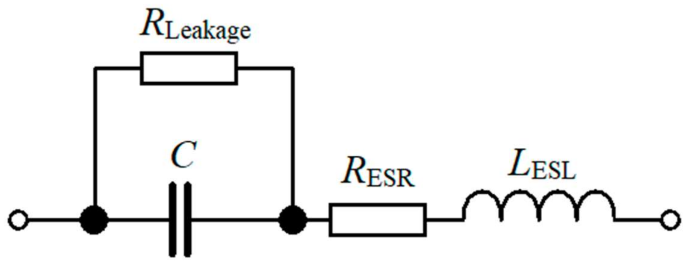


30. Draw a Realistic Circuit Model for a Capacitor. What Are the Parasitics and Where Do They Come From?

In real-world capacitors, parasitic elements include:

Equivalent Series Resistance (ESR): Due to internal resistance of the capacitor materials.
Parasitic Inductance: Due to the leads and connections.

Realistic model:



C is the ideal capacitance,
Resr is the equivalent series resistance,
LESL is the parasitic inductance.

31. What is a Capacitor's Self-Resonant Frequency?

The self-resonant frequency (SRF) is the frequency at which the capacitor’s inductive and capacitive reactances cancel each other out. Above this frequency, the capacitor behaves like an inductor due to the dominance of parasitic inductance.

32. What is a Bypass/Decoupling Capacitor? What About a Bulk Capacitor? What is the Difference?

Bypass/Decoupling Capacitor: A small capacitor placed near an IC to filter high-frequency noise from the power supply. It ensures smooth operation by decoupling the IC from the power supply noise.

Bulk Capacitor: A larger capacitor used to store energy and provide it when there’s a dip in the power supply. It stabilizes the power supply at lower frequencies.

Difference:

Bypass capacitors filter out high-frequency noise.
Bulk capacitors smooth out low-frequency variations and store energy.

33. What is an AC Coupling Capacitor?

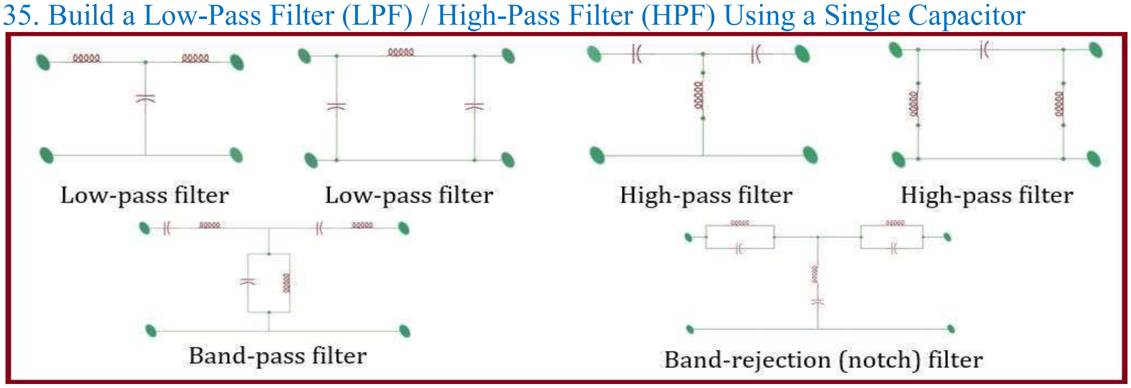
An AC coupling capacitor blocks DC signals while allowing AC signals to pass through. It’s used in circuits to remove unwanted DC offsets from the signal.

34. What Sort of Signals Can Capacitors Pass Through and Block? What Sort of Filter Behavior is This?

Capacitors pass AC signals: They allow high-frequency signals to pass and block low-frequency signals.
Capacitors block DC signals: They prevent the flow of direct current after being charged.

This behavior is characteristic of a high-pass filter (HPF).

35. Build a Low-Pass Filter (LPF) / High-Pass Filter (HPF) Using a Single Capacitor



Low-Pass Filter (LPF):
Place the capacitor in parallel with the output.
High-Pass Filter (HPF):

Place the capacitor in series with the input.

The cutoff frequency fc is: $fc=1/2\pi RC$

36. What Are Some Common Failure Modes of a Capacitor?

Short Circuit: The dielectric fails, causing the capacitor to act like a wire.
Open Circuit: The internal connections break, causing no capacitance.
ESR Increase: The equivalent series resistance increases, reducing the capacitor’s efficiency, especially in high-frequency applications.
Leakage Current: Over time, the capacitor may start leaking current, reducing its effectiveness.

37. What is the Continuity Condition? What Do Capacitors Resist Change to?

The continuity condition for a capacitor is that the current through a capacitor can change instantaneously, but the voltage cannot.

Capacitors resist changes in voltage. They take time to charge or discharge, which is why they smooth voltage fluctuations in circuits.

Inductors

38. Draw the Symbol for an Inductor

The symbol for an inductor is a coil of wire:

___/\/\/\/___

39. What is Inductance?

Inductance is the property of an inductor that opposes changes in current. It stores energy in a magnetic field when current flows through it, measured in Henrys (H).

40. What is the Equation for Impedance of an Inductor?

The impedance ZL of an inductor is given by: $ZL=j\omega L$ Where:

J is the imaginary unit,
$\omega=2\pi f$ is the angular frequency,
Lis the inductance.

41. What is the Differential Equation for an Inductor?

The voltage V across an inductor is proportional to the rate of change of current I: $V(t)=LdI(t)/dt$

42. Does an Inductor Have Positive or Negative Reactance?

An inductor has positive reactance, meaning it resists changes in current as frequency increases: $XL=\omega L$

43. What is the Continuity Condition? What Do Inductors Resist Change to?

Inductors resist sudden changes in current. The current through an inductor cannot change instantaneously.

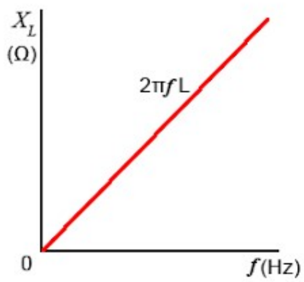
44. How Does an Inductor Behave When Initially Excited and at DC Steady-State?

Initially excited: The inductor resists current changes, causing the current to ramp up gradually.
At DC steady-state: The inductor acts like a short circuit (zero impedance) at DC.

45. What is an Inductor's Impedance at DC vs Infinitely High Frequency? Plot This Behavior.

At DC: The impedance is zero ($ZL=0$).
At high frequency: The impedance becomes very large ($ZL\to\infty$).

Plot:



46. What Happens When an Inductor Saturates?

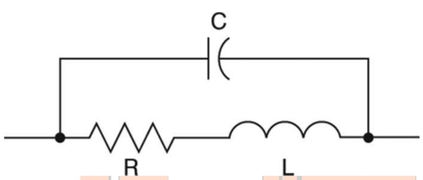
When an inductor saturates, its core material can no longer store additional magnetic energy, and its inductance decreases drastically, reducing its ability to oppose changes in current.

47. Draw a Realistic Circuit Model for an Inductor. What Are the Parasitics and Where Do They Come From?

Real-world inductors have parasitic components like:

Parasitic Resistance (R_esr): Due to the wire's resistance.
Parasitic Capacitance (C_parasitic): Due to the winding's turns being close together.

Realistic model:



48. What Do Cores Do on Inductors?

Cores in inductors increase the inductance by concentrating the magnetic field. Cores also improve efficiency by reducing magnetic losses.

49. What Are Typical Core Materials?

Common core materials include:

Ferrite (for high frequencies),
Iron powder,
Laminated steel (for power applications).

50. What Are the Main Loss Mechanisms of an Inductor? Where Do They Arise From?

The main loss mechanisms in an inductor are:

Core Losses: Arise from the magnetic material of the core, including hysteresis and eddy current losses.
Hysteresis Loss: Caused by the continuous magnetization and demagnetization of the core as the current changes.
Eddy Current Loss: Induced currents in the core material itself due to changing magnetic fields, leading to heat dissipation.
Copper Losses (I^2R): Arise from the resistance of the wire windings, causing power loss as heat when current flows through them.
Leakage Inductance: Part of the magnetic flux leaks outside the core, reducing efficiency.
Parasitic Capacitance: Due to the proximity of the winding turns, which can lead to energy losses, especially at high frequencies.

51. What is ACR? Where Does It Arise From and How Is It Impacted by Frequency?

ACR (AC Resistance) is the effective resistance of an inductor at AC (alternating current) frequencies. It arises from:

Skin Effect: At higher frequencies, current tends to flow on the surface of the conductor, increasing resistance.
Proximity Effect: At high frequencies, current flow in nearby conductors induces eddy currents, further increasing resistance.

As the frequency increases, both skin effect and proximity effect become more pronounced, leading to a higher ACR.

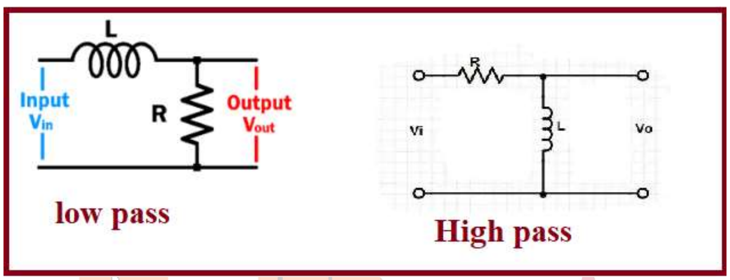
52. What Sort of Signals Can Inductors Pass Through and Block? What Sort of Filter Behavior Is This?

Inductors pass low-frequency (including DC) signals: Because the impedance is low at low frequencies.
Inductors block high-frequency signals: Due to their increasing impedance with frequency.

This behavior is characteristic of a low-pass filter (LPF), where low-frequency signals pass through and high-frequency signals are blocked.

53. Build a Low-Pass Filter (LPF) / High-Pass Filter (HPF) Using a Single Inductor

Low-Pass Filter (LPF): Place the inductor in series with the input.



The cutoff frequency fc is: fc=R/2πL
High-Pass Filter (HPF): Place the inductor in parallel with the output.

The cutoff frequency fcis the same formula: fc=R/2πL

54. What Are Some Common Failure Modes of an Inductor?

Common failure modes of inductors include:

Open Circuit: The wire inside the inductor breaks, causing the inductor to stop conducting.
Short Circuit: Insulation between windings breaks down, causing a short.
Saturation: The core material saturates and the inductance drops, leading to poor performance under high current.
Overheating: Excessive current causes heating, which can lead to wire insulation breakdown or even melting.
Core Losses: Magnetic core material deteriorates or loses efficiency, especially at high frequencies.

Physics/Chemistry

Basic physics/chemistry questions, not including device physics or basic circuit physics which have their own sections.

55. What is a Conductor/Insulator/Semiconductor?

Conductor: A material that allows electricity to flow easily (e.g., copper, aluminum). It has many free electrons.

Insulator: A material that resists the flow of electricity (e.g., rubber, plastic). It has few or no free electrons.

Semiconductor: A material that can conduct electricity under certain conditions (e.g., silicon). Its conductivity lies between conductors and insulators and can be controlled.

56. What is the Skin Effect?

The skin effect is a phenomenon where AC (alternating current) flows primarily near the surface of a conductor at high frequencies, reducing the effective area for current flow and increasing resistance.

57. What is the Photoelectric Effect?

The photoelectric effect occurs when light shines on a material (usually metal) and causes it to emit electrons. This phenomenon is the basis for devices like solar cells.

58. Explain Any Dielectric Losses and How They Occur.

Dielectric losses occur in insulating materials (dielectrics) when they are subjected to a changing electric field. Energy is lost as heat due to the lag (hysteresis) in the polarization of the dielectric material, especially at high frequencies.

59. How Does Electricity Work?

Electricity is the flow of electric charge, usually carried by electrons moving through a conductor (like a wire). It is driven by a difference in voltage (potential difference) and flows from higher to lower potential.

60. How Does Lightning Work?

Lightning occurs when electrical charges build up in clouds due to the movement of water droplets. The difference in charge between the cloud and the ground or between clouds causes a rapid discharge of electricity, creating a lightning bolt.

61. Which Atom Bands Do Electrons Move In?

Electrons move in energy levels or bands around the nucleus of an atom:

Valence Band: The outermost band where electrons are involved in bonding.

Conduction Band: The next higher band where electrons are free to move and conduct electricity. In conductors, the conduction band overlaps the valence band, while in semiconductors, they are close but separate, and in insulators, they are far apart.

Semiconductor Devices

Device Physics

62. What is the Difference Between P-Type and N-Type Semiconductors?

P-Type: Has an excess of holes (positive charge carriers) due to doping with elements like boron.
N-Type: Has an excess of electrons (negative charge carriers) due to doping with elements like phosphorus.

63. What is a Depletion Region and How Does It Form?

The depletion region forms at the junction between P-type and N-type semiconductors in a diode. It forms when electrons from the N-side combine with holes from the P-side, leaving behind charged ions and creating a region with no free charge carriers.

64. What Materials Are Typically Used for Semiconductors?

Common materials for semiconductors include:
Silicon (Si)
Germanium (Ge)
Gallium arsenide (GaAs)

65. What is Doping?

Doping is the process of adding small amounts of impurities to a semiconductor to change its electrical properties by increasing the number of free charge carriers (electrons or holes).

66. How Does Doping Improve Performance?

Doping improves performance by increasing the number of free charge carriers (electrons or holes), allowing the semiconductor to conduct electricity more easily.

67. What Elements Are Typically Used for Doping?

N-Type doping: Uses elements like phosphorus or arsenic (which donate electrons).
P-Type doping: Uses elements like boron or gallium (which create holes).

68. What is Charge Mobility?

Charge mobility is the ability of charge carriers (electrons or holes) to move through a semiconductor material when an electric field is applied.

69. What Charge Carriers Are Dominant in P-Type vs N-Type Semiconductors?

P-Type: Dominant charge carriers are holes (positive charges).
N-Type: Dominant charge carriers are electrons (negative charges).

70. Which Charge Carriers Are More Mobile: Holes or Electrons?

Electrons are more mobile than holes because they are lighter and can move more easily through the crystal lattice.

71. Why Do N-Type Semiconductors Typically Have Lower Resistance?

N-Type semiconductors have lower resistance because electrons, which are the charge carriers, are more mobile and can move more easily compared to holes in P-Type semiconductors.

Diodes

72. How Does a Diode Work?

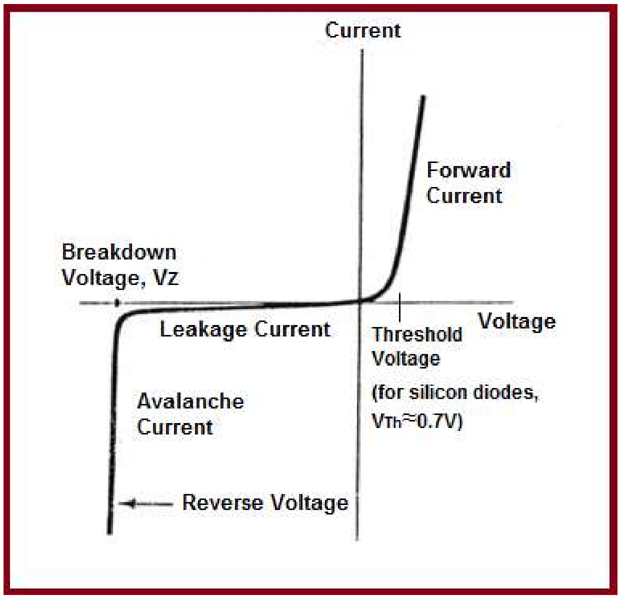
A diode allows current to flow in only one direction (from anode to cathode). When forward biased (positive voltage at anode), it conducts; when reverse biased, it blocks current.

73. What Are Some Use Cases of a Diode?

Rectification: Converting AC to DC.
Clamping/Clipping: Limiting voltage levels.
Switching: Used in logic circuits.
Protection: Preventing reverse current in circuits.

74. Draw IV Curve

The IV curve of a diode looks like this:



In forward bias, current increases exponentially after the threshold voltage. In reverse bias, current is nearly zero until breakdown.

75. What Are the Different Operation Regions?

Forward Bias: Diode conducts once the forward voltage exceeds the threshold.
Reverse Bias: Diode blocks current, except a small leakage current.
Reverse Breakdown: At high reverse voltage, the diode conducts in reverse.

76. Given a Forward/Reverse Voltage and Threshold Voltage, Determine What Operation Region It's In

Forward Bias: If the applied forward voltage > threshold voltage (usually ~0.7V for silicon).
Reverse Bias: If the applied reverse voltage is less than breakdown voltage.
Reverse Breakdown: If the reverse voltage exceeds the breakdown threshold.

77. What Are the Different Kinds of Diodes?

Standard Diode: For rectification.
Zener Diode: For voltage regulation.
Schottky Diode: Low forward voltage drop.
Light Emitting Diode (LED): Emits light when forward biased.
Photodiode: Converts light into current.

78. What Is the Typical Forward Voltage Drop Across a Diode?

Silicon Diode: ~0.7V forward voltage drop.
Schottky Diode: ~0.2V-0.3V (lower than silicon).
LED: Varies (1.8V-3.3V), depending on color.

This value is due to the material properties and the energy needed for electrons to cross the junction.

79. How Can You Build an Ideal Diode?

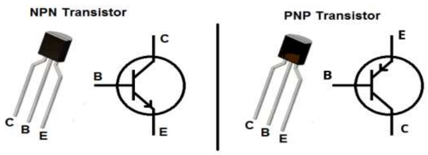
An ideal diode would have:

Zero forward voltage drop.
Infinite reverse resistance (no reverse current). In practice, this isn't possible, but circuits like ideal diode controllers use MOSFETs to mimic ideal behavior.

BJTs

BJTs (Bipolar Junction Transistors)

80. Draw a NPN/BJT Symbol



81. What Are the Terminals?

C: Collector
B: Base
E: Emitter

82. What Are Some Use Cases of a BJT?

Switching: Used to turn circuits on/off.
Amplification: Amplifies weak signals in audio or RF circuits.

83. What Is the Difference Between a BJT and MOSFET?

BJT: Current-controlled (base current drives the transistor).
MOSFET: Voltage-controlled (gate voltage drives the transistor).

84. Draw IV Curve

In active region, current increases exponentially with base-emitter voltage.
Saturation: When the transistor is fully on.
Cutoff: When the transistor is off.

85. How Does It Work?

Small current at the base controls a larger current between the collector and emitter (for NPN: current flows from collector to emitter when base is positive).

86. What Is the Difference Between an NPN and PNP?

NPN: Current flows from collector to emitter when base is positive.
PNP: Current flows from emitter to collector when base is negative.

87. What Are the Different Modes of Operation?

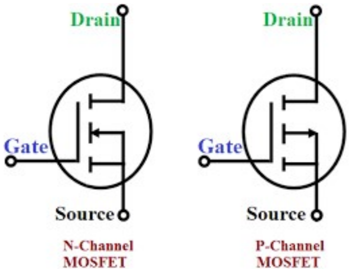
Cutoff: BJT is off, no current flows.
Active: BJT amplifies, collector current controlled by base current.
Saturation: BJT is fully on, acting as a closed switch.

88. What Is Current Gain/Beta?

Beta (β) is the ratio of collector current to base current: $\beta = I_C/I_B$

MOSFETs

89. Draw a PFET/NFET Symbol



90. What Are the Terminals?

D: Drain
G: Gate
S: Source

91. Which Two Terminals Are Often Connected Together and Why?

Source and body (or bulk) are often connected to simplify circuit design and prevent body effect (affecting threshold voltage).

92. What Are Some Use Cases of a MOSFET?

Switching: Power supplies, motor control.
Amplification: Used in RF and audio applications.

93. Why Are MOSFETs Used So Often?

Low power consumption (voltage-controlled).
Fast switching speeds.
High efficiency in power electronics.

94. What Is the Difference Between a BJT and MOSFET?

BJT: Current-controlled.
MOSFET: Voltage-controlled.

95. Draw IV Curve for MOSFET

The linear region is where current increases linearly with voltage.
The saturation region is where current stays constant despite increasing voltage.

96. How Does It Work?

A voltage applied at the gate controls the flow of current between the drain and source (N-channel: current flows when gate is positive).

97. What Is the Difference Between N-Channel and P-Channel?

N-Channel: Current flows when gate is positive.
P-Channel: Current flows when gate is negative.

98. What Is the Difference Between Enhancement-Mode and Depletion-Mode?

Enhancement-mode: MOSFET is off at zero gate voltage, needs a positive gate voltage (N-channel) to turn on.
Depletion-mode: MOSFET is on at zero gate voltage, can be turned off by applying voltage.

99. What Are the Different Modes of Operation?

Cutoff: No current flows (off).
Linear (Ohmic): MOSFET behaves like a resistor.
Saturation: MOSFET fully on, acts as a switch.

100. Given Vgs and Vt, How Will the FET Behave?

If $V_{gs} > V_{th}$ (threshold voltage): MOSFET is on (linear or saturation).
If $V_{gs} < V_{th}$: MOSFET is off (cutoff).

101. What Is the Internal Body Diode? Where Does It Come From?

The body diode exists between the drain and source due to the p-n junction in MOSFETs. It provides protection against reverse current.

102. Why Are MOSFETs Sensitive to Overvoltage Conditions at the Gate?

The gate oxide layer is very thin, and excessive voltage can break it down, permanently damaging the MOSFET.

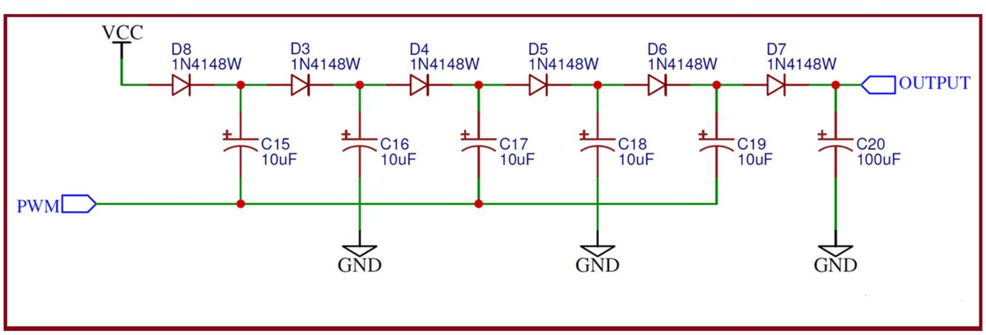
103. What Is W/L Ratio? How Does It Impact FET Performance?

W/L is the ratio of the channel width to length. A higher W/L ratio increases current-carrying capability, improving performance and reducing resistance.

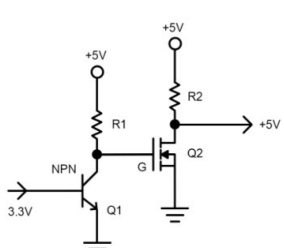
Analog Circuits/Electronics

104.Often times, I'm given an unnamed circuit and asked to explain how it works and what type of circuit it is. Here are a few of them I've gotten (and know what they're called):

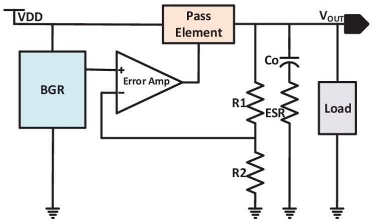
- Charge pump



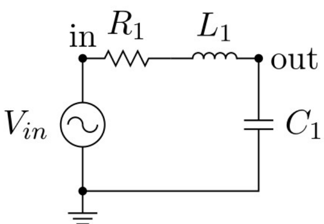
- Level shifter



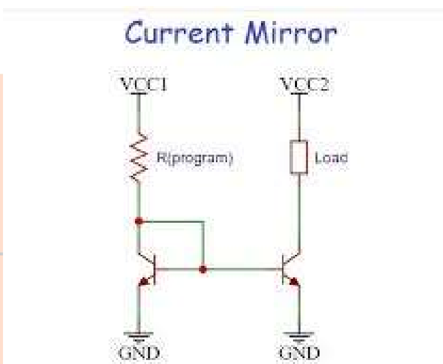
- LDO



- RLC filters



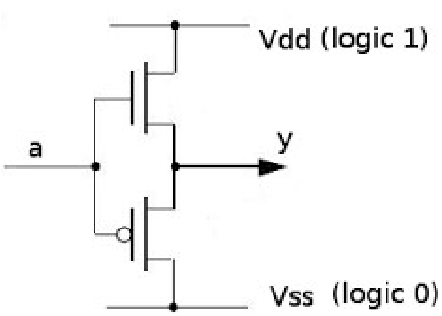
- Current mirror



CMOS

105. What Is CMOS? Draw a CMOS Buffer.

CMOS: Technology using complementary pairs of p-type and n-type MOSFETs.
CMOS Buffer:



106. Why Are These So Often Used?

Low power consumption: Only consumes power when switching.
High noise immunity.

107. What Type of MOSFETs Are Typically Used on the High-Side? Low-Side? Why?

High-Side: PFET (simpler drive circuitry for pulling up).
Low-Side: NFET (better performance for pulling down, higher current capacity).

108. TTL vs CMOS?

TTL: Faster, higher power consumption.
CMOS: Slower, lower power consumption, better for large scale integration.

109. What Is Shoot-Through on a CMOS Logic Gate? How Can It Be Prevented?

Shoot-through occurs when both PFET and NFET conduct briefly during switching, causing a short. Prevent it with careful timing or adding small delay circuits.

110. CMOS Efficiency Analysis: Impact of Frequency, Logic Levels, Rise/Fall Times?

Higher frequency: More dynamic power consumption.
Logic swing: Larger swings increase power.
Faster rise/fall times: Reduce power loss.

111. How Can You Achieve Faster Slew Rates?

Increase W/L ratio of the MOSFETs.
Use higher current drive.

112. Design CMOS Input Protection Circuits Against ESD/Overvoltage Events.

Use clamping diodes and series resistors at inputs.
Add TVS (transient voltage suppression) diodes for additional protection.

113. Voltage/Current Output Waveforms for CMOS Buffer with RC Load?

During a positive step, the NFET conducts.
During a negative step, the PFET conducts.
Waveform: Exponential rise/fall due to the RC load.

114. What Happens If You Swap the High-Side PFET and Low-Side NFET?

The circuit won't work properly because the NFET requires a gate voltage higher than the source to turn on, and vice versa for PFET.

Amplifiers

115. What Is Common-Mode and Differential Gain?

Common-mode gain: Gain for signals common to both inputs, ideally zero.
Differential gain: Gain for the difference between inputs, ideally very high.

116. What Is the Gain-Bandwidth Product?

The product of the amplifier's gain and bandwidth. Higher gain reduces bandwidth and vice versa.

117. What Determines Slew Rate and Rise/Fall Time?

Determined by the capacitive load and the current drive capability of the amplifier.

118. Pros/Cons of Fast/Slow Slew Rates?

Fast: Better performance, less distortion at high frequencies.
Slow: Reduced risk of instability, less ringing.

119. How to Bias an Amplifier?

Set a quiescent operating point by selecting appropriate resistors or current sources in the circuit.

120. Why Are BJTs More Often Used for Amplifiers than MOSFETs?

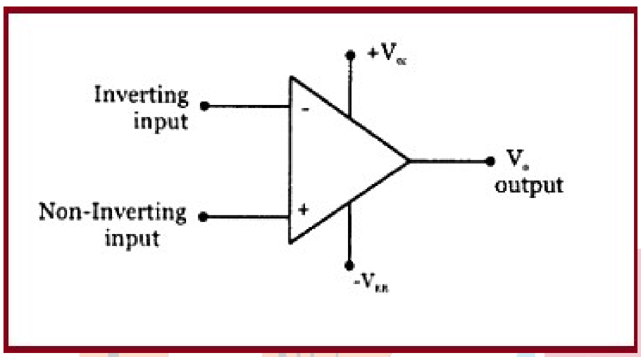
BJTs have higher gain and better linearity for analog signals, especially at lower currents.

Opamps (Operational Amplifiers)

121. What Is an Opamp? How Does It Work?

An opamp is an amplifier with very high gain used in feedback circuits to perform mathematical operations like addition, subtraction, integration, etc.

122. Draw an Opamp. What Are the Two Input Terminals?



Two inputs: Inverting (-), Non-inverting (+).

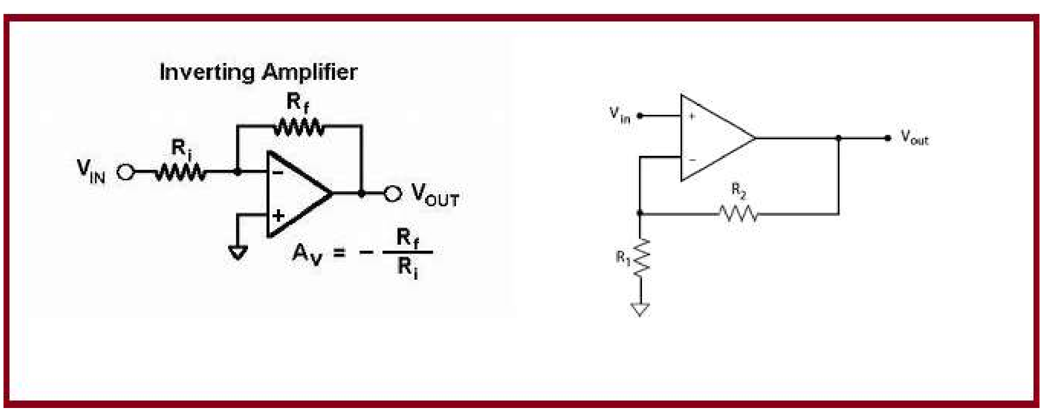
123. What Connections Are Required to Wire Up an Opamp?

Power supply (Vcc and GND).
Input signals to inverting/non-inverting terminals.
Feedback from output to input.

124. What Are the Three Rules of an Ideal Opamp?

Infinite gain.
Infinite input impedance.
Zero output impedance.

125. Draw an Inverting/Non-Inverting Opamp Circuit



126. What Is the DC Gain of an Opamp?

For an inverting amplifier: −Rf/Rin
For a non-inverting amplifier: 1+(Rf/Rin).

127. How Does an Open-Loop Opamp Behave? What Type of Circuit Is It?

In open-loop, the opamp has very high gain and is used in comparator circuits. Any small input difference drives the output to the supply rails.

128. Why Are Non-Inverting Configurations Often Preferred?

Non-inverting configurations offer high input impedance, minimizing signal loading.

Filters

129. Draw Mag/Phase Response of a First-Order Filter

For a low-pass filter: Magnitude decreases, phase shifts negative as frequency increases.

130. Draw a Circuit for a First-Order LPF/HPF

LPF: Resistor in series with input, capacitor to ground at output.
HPF: Capacitor in series with input, resistor to ground at output.

131. What Is the Dropoff Rate of a First-Order Low-Pass Filter?

The dropoff rate is 20 dB/decade beyond the cutoff frequency.

132. What Does the Bandwidth Characterize?

The bandwidth characterizes the range of frequencies over which the filter passes signals without significant attenuation.

133. What Is the 3dB Rolloff Point?

At the 3dB rolloff, the output signal is reduced to 70.7% of its original value. Beyond this, the filter attenuates more sharply.

PLL

134. What Can You Use to Increase Clock Frequency?

Use a Phase-Locked Loop (PLL) or frequency multiplier circuits to increase the clock frequency.

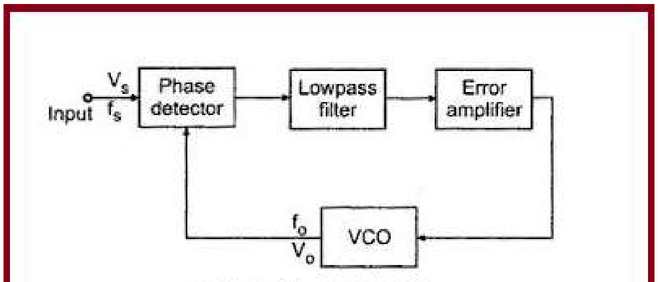
135. How Does a PLL Work?

A Phase-Locked Loop (PLL) adjusts the phase and frequency of its output to match an input signal, locking onto it and stabilizing the output frequency.

136. What Blocks Does a PLL Have?

Phase Detector (PD): Compares input and output phase.
Low-Pass Filter (LPF): Filters out high-frequency noise.
Voltage-Controlled Oscillator (VCO): Adjusts the frequency based on feedback.
Frequency Divider (optional): Scales frequency to desired value.

137. Draw a Block Diagram



Block diagram of PLL

138. What Happens If Your PLL Locks Too Fast or Too Slow?

Too fast: Causes instability or jitter.
Too slow: PLL may fail to lock or track input changes accurately.

139. Why Is Crushing the Clock Window Bad?

Crushing the clock window (reducing timing margin) can cause timing violations in connected circuits, leading to malfunction, data corruption, or missed timing requirements.

Power Electronics

140. What type of load can a SoC/CPU typically be characterized as?

SoC/CPU loads are typically capacitive, due to the high number of switching transistors.

141. What is the condition for maximum power transfer?

Maximum power transfer occurs when the load resistance equals the source resistance (R_load = R_source).

142. Why are there many capacitors connected to ground on a power rail?

These are decoupling capacitors used to stabilize the voltage by filtering out noise. Multiple capacitors with different values provide filtering over a wider range of frequencies. One massive capacitor would be slower to respond to high-frequency changes.

143. What is bulk capacitance?

Bulk capacitance is used to store energy and stabilize the power supply, especially during large load changes.

144. What is inductive flyback?

Inductive flyback is the high voltage generated when current through an inductor is suddenly interrupted. It's unwanted in switching circuits (can damage components) but wanted in flyback converters. Protection: use diodes (flyback or freewheeling diodes).

145. What is PWM?

Pulse Width Modulation (PWM) is a technique where the duty cycle of a signal controls the average voltage or power delivered to a load.

146. What novel semiconductors are being explored for power electronics?

Silicon Carbide (SiC) and Gallium Nitride (GaN). Tradeoffs: SiC and GaN offer higher efficiency and faster switching but are more expensive.

General Power Supply

147. What are some ways to step up/down voltage?

Step-up: Boost converter, transformer.
Step-down: Buck converter, LDO, transformer.

148. Why shouldn't you use a voltage divider for a power rail?

A voltage divider can’t handle variable loads or large current without power loss, making it inefficient.

149. When is a voltage divider applicable to step down voltage?

For low-current applications like signal conditioning, not power rails.

150. Design a power supply.

Example: A buck converter design with a switching regulator, inductor, diode, and output capacitor to step down voltage efficiently.

151. Why is supply voltage overshoot/undershoot undesired?

It can damage sensitive loads like CPUs and memory chips. It's tolerated in less sensitive circuits but generally avoided.

152. What parameters would you want to track in a power supply?

Input/output voltage, input/output current, power efficiency, temperature, and ripple.

153. What specs do you look out for when buying/designing a power supply?

Efficiency, ripple, voltage regulation, load regulation, current capacity, and protection features (overvoltage, overcurrent).

154. Why is energy conservation important?

It improves efficiency, reduces waste, saves costs, and minimizes heat generation.

155. Calculate the fourth value: input voltage, input current, output voltage, output current.

Use Power In = Power Out: $V_{in} \times I_{in} = V_{out} \times I_{out}$.

Power Supply Topologies

156. Name some voltage regulators.

Linear regulators (LDO), buck converters, boost converters, buck-boost converters.

157. Difference between buck converter vs LDO?

Buck is a switching regulator, efficient for large voltage drops. LDO is simpler but less efficient for high voltage differences.

158. When can an LDO be more efficient than a buck converter?

When the input-output voltage difference is small, an LDO can be more efficient due to no switching losses.

159. What circuits convert AC-AC/AC-DC/DC-DC/DC-AC?

AC-AC: Transformer.
AC-DC: Rectifier.
DC-DC: Buck/boost converter.
DC-AC: Inverter.

160. What is a bridge rectifier?

A bridge rectifier converts AC to DC using four diodes to ensure current flows in one direction.

161. Why are step-down regulators more common in embedded systems?

Most embedded systems use low voltage, so step-down (buck) converters are more needed than step-up.

Efficiency

162. What is efficiency in a power supply?

Efficiency = PoutPin×100%PinPout×100%. It's important to minimize losses and heat generation.

163. Calculate the third value: input power, output power, efficiency.

Efficiency=PoutPin×100%Efficiency=PinPout×100%, rearrange to solve for the missing variable.

164. What happens to lost power?

Lost power turns into heat, which can damage components and reduce efficiency.

165. What is quiescent current?

Quiescent current is the small current drawn by a power supply when no load is connected. It affects overall efficiency.

Power Architecture

166. Design power architecture given supply specs.

Start with DC-DC converters (buck, boost) or LDOs to step down/up voltages. Ensure the output current rating matches the load requirements. Add decoupling capacitors for stability.

167. Solutions for multiple output supplies from a single input.

Use a multiple-output converter or SIMO (Single Input Multiple Output) converter, or cascade multiple buck/boost converters from the same input.

168. Solutions for multiple output supplies with different noise performance.

Use dedicated regulators (e.g., separate LDOs) for each output or use filtering techniques to clean the noise.

169. Solution for high dropout voltage with high efficiency and clean output.

Use a buck converter for efficiency and follow it with an LDO for a clean, stable output.

170. What is an intermediate bus converter?

It steps down a higher input voltage to an intermediate level before distributing it to point-of-load converters. Needed in distributed power architectures for efficiency.

171. What is power sequencing and when is it needed?

Power sequencing is the timed turning on/off of power rails in a specific order. It's needed to ensure proper startup/shutdown of components. The sequence depends on component requirements, with gaps of milliseconds between rails.

172. How to implement power sequencing?

Use sequencing ICs, microcontrollers, or discrete circuits like comparators and capacitors to control the timing of each power rail.

Buck Converters

173. What is a buck converter?

A buck converter steps down DC voltage efficiently using a switch, inductor, diode, and capacitor.

174. How does a buck converter work?

During the on state, the switch connects the input to the inductor, storing energy. In the off state, the inductor releases stored energy to the output through the diode or low-side FET.

175. Draw a circuit for a buck converter.

A buck converter circuit includes a switching transistor (FET), inductor, diode, and output capacitor.

176. Applications of buck converters.

Used in power supplies for CPUs, GPUs, mobile devices, and battery chargers.

177. What is a SIMO buck converter?

A Single Input Multiple Output buck converter delivers multiple output voltages from a single input by time-sharing the inductor.

Duty Cycle/Output Voltage

178. What determines the output voltage in a buck converter?

Duty cycle and input voltage determine output voltage. Vout=Vin×Duty CycleVout=Vin×Duty Cycle.

179. What is duty cycle?

Duty cycle is the ratio of the on-time of the switch to the total time of the switching cycle.

180. How to calculate duty cycle?

Duty Cycle=Vout/Vin.

181. What happens at 100% and 0% duty cycle?

100%: Output equals input.
0%: Output is zero.

182. Given Vout/Vin ratio, how to calculate duty cycle?

Duty Cycle=Vout/Vin

183. Given duty cycle and Vin, how to find Vout?

Vout=Vin×Duty Cycle.

184. What is a DC load-line?

A graphical representation of current vs. voltage. It helps in understanding the operating points of a circuit, but it can limit flexibility in dynamic systems.

Circuit Analysis

185. Draw waveforms for various components in a buck converter.

- Inductor voltage/current
- Switch node voltage under asynchronous and synchronous rectification
- Capacitor voltage/current
- FET gate voltage
- Diode current/voltage
- Input supply voltage/current
- Output load voltage/current
- Low-side NFET voltage/current if using synchronous rectification

186. Purpose of output inductor/capacitor.

Inductor stores energy, and capacitor filters and smooths the output voltage.

187. Current direction in inductor/capacitor during on/off states.

On state: Inductor charges, capacitor supplies load. Off state: Inductor discharges, capacitor continues to supply load.

188. Why is inductor current a linear ramp?

The voltage across the inductor changes at a constant rate, resulting in a linear current ramp. The slope is determined by VL=Ldi/dt.

189. Why is the output capacitor voltage a linear ramp?

The capacitor charges/discharges as current flows, creating a linear voltage change based on current and capacitance.

190. When does the output capacitor charge/discharge?

Charges during on state when the inductor is supplying energy. Discharges during off state to maintain output voltage.

191. What is the inductor polarity?

The polarity depends on current flow, flipping when switching between on and off states.

192. When does the inductor sink/supply current?

The inductor sinks current during the on state and supplies current during the off state.

193. What is the input capacitor needed for?

The input capacitor smooths out input voltage ripples. Without it, input voltage can oscillate, causing instability.

194. What is voltage/current ripple?

Ripple is the fluctuation in voltage/current caused by the switching of the converter. Minimizing ripple is key for stability.

195. What determines inductor current slope?

Inductor current slope is determined by the voltage across the inductor and the inductance value.

196. What determines capacitor voltage slope?

The capacitor voltage slope is determined by the current through the capacitor and its capacitance.

197. Why is the diode needed?

The diode provides a path for inductor current during the off state. Removing it would break the current flow.

198. Impact of replacing the diode with an ideal diode.

Replacing with an ideal diode improves efficiency by eliminating forward voltage drop losses.

199. Switch node voltage potential in off state.

In the off state, the switch node is at ground or a negative value depending on the circuit.

200. Where is inductor current sourced from in off state?

The inductor current is sourced from the diode or low-side FET in synchronous rectification.

Power Bridge

Many of these questions assume that synchronous rectification is implemented, but some also compare the behavior/performance between a low-side NFET and diode.

201. What type of power bridge is often implemented? Why?

H-bridge is common, used to control direction of current in motors and provide bidirectional control.

202. Why is the diode needed? What if it's removed?

The diode blocks reverse current and provides freewheeling for the inductor. Without it, the circuit could fail due to back EMF or large voltage spikes.

203. Explain synchronous rectification. What are its pros/cons?

Synchronous rectification replaces diodes with FETs to reduce conduction losses.

Pros: Higher efficiency.

Cons: More complex control circuitry and potential for shoot-through.

204. NFET vs PFET for high-side FET? What are pros and cons?

NFET: Lower Rds(on), better efficiency, but needs a bootstrap or charge pump to drive.

PFET: Easier to drive but has higher Rds(on) and lower efficiency.

205. What are the pros and cons of a symmetrical power bridge?

Pros: Equal distribution of current, balanced thermal dissipation.

Cons: More components and complexity.

206. Design a circuit to properly drive a high-side NFET. What considerations must be taken into account?

Use a bootstrap circuit or charge pump for gate drive voltage.

Ensure sufficient gate drive voltage above source and prevent shoot-through.

207. What happens if both the NFET and PFET are on at the same time? How do you avoid this?

It causes a short circuit through the supply. Avoid this by adding dead time between switching the FETs.

208. What happens to the internal body diode in the off state? Draw the switch node voltage waveform.

The body diode conducts when the FET is off if there's a voltage across it, causing reverse current flow.

209. What considerations must be taken given the internal body diode behavior?

Ensure synchronous rectification to reduce losses from the body diode's forward voltage drop.

210. What can you do to decrease the effect of the internal body diode?

Use synchronous rectification (replace diodes with active switches like MOSFETs).

Efficiency, Ripple, and Switching Frequency

211. Why are buck converters more efficient than LDOs?

Buck converters use switching regulation, wasting less energy as heat, while LDOs dissipate excess voltage as heat.

212. How to size inductors/capacitors to meet ripple spec?

Calculate inductor value for desired ripple current and capacitor value for ripple voltage, using formulas based on switching frequency and load current.

213. Component-by-component analysis on efficiency and ripple impact.

Inductor: Impacts current ripple and losses (core and winding).

Capacitor: Affects voltage ripple.

FETs: Switching and conduction losses impact efficiency.

Diode: Adds conduction loss and reverse recovery.

214. What are some ways to increase efficiency?

Use low Rds(on) FETs, minimize switching losses, use synchronous rectification, and optimize inductor sizing.

215. What are some ways to decrease ripple?

Increase inductance or capacitance, increase switching frequency, and use low ESR capacitors.

216. How does switching frequency impact efficiency/ripple?

Higher frequency reduces ripple, but increases switching losses, lowering efficiency.

217. What components are most responsible for efficiency losses?

FETs (switching/conduction losses), inductor (core/winding losses), and diode (forward voltage drop).

218. Why should ripple be decreased?

Excessive ripple can cause noise, instability, and damage to sensitive components.

219. When does ripple not matter as much?

Ripple is less critical in robust loads like motors and heaters, which aren't sensitive to minor fluctuations.

220. What limits the switching frequency on the upper and lower ends?

Upper limit: Parasitic capacitance and inductance cause switching losses and heat.

Lower limit: Ripple becomes too high, and the converter may become unstable.

221. How does switching frequency impact performance?

Higher frequency can improve transient response and reduce ripple but at the cost of efficiency.

222. How does dropout voltage impact efficiency, ripple, etc.?

Higher dropout voltage reduces efficiency and may increase ripple due to higher current demands.

Voltage/Current Sensing

223. How is the output voltage typically sensed?

Via a resistor divider network, which feeds back to the control loop to regulate voltage.

224. What happens if the voltage-sensing resistor divider has too high resistance?

It may cause signal distortion and slower response times.

Too low: Excessive current draw from the feedback loop.

225. How can the output current be sensed?

Using a current-sense resistor or Hall effect sensor.

226. When should output current be sensed?

After the output inductor for accurate DC current measurement, or before for total system current.

227. Which voltages/currents are important in the control loop?

Output voltage for regulation, input current to manage efficiency, and inductor current for stability.

PFM Operation

228. What are the different operation modes?

Continuous and discontinuous conduction modes, PFM (pulse-frequency modulation), and PWM.

229. What is the difference between discontinuous and continuous operation?

Continuous: Inductor current never reaches zero.

Discontinuous: Inductor current reaches zero each cycle.

230. What can you do to increase efficiency at light loads?

Use PFM instead of PWM, which adjusts frequency based on load, reducing switching losses.

231. Explain PFM operation.

Pulse-Frequency Modulation lowers switching frequency at light loads to save power.

232. What is the difference between PFM and PWM?

PFM adjusts frequency based on load, while PWM keeps a constant frequency but adjusts the duty cycle.

233. How does the ripple compare between PFM and PWM?

PFM typically has higher ripple due to fewer switching cycles, while PWM maintains lower, consistent ripple.

234. What happens to the inductor current?

It ramps up when the switch is on and down when the switch is off.

235. Describe the charge/discharge cycle of inductor and capacitor.

Inductor charges during the on-state, stores energy, and discharges during the off-state into the capacitor, which smooths the output voltage.

Multiphase Buck Converters

236. What is a multiphase buck converter?

A buck converter with multiple phases to share current load.

237. What are the pros/cons vs single-phase buck converters?

Pros: Higher efficiency, lower ripple, better heat distribution.

Cons: More complex, larger size.

238. How do they work and what challenges are presented?

Multiple phases switch out of phase to balance load, but require phase balancing and more control circuitry.

239. When should they be used?

For high-current applications or when thermal management is critical.

240. How do they increase output current?

By distributing current across multiple phases, reducing stress on individual components.

241. How do they impact efficiency?

Increase efficiency by reducing switching and conduction losses.

242. How do they impact heat density?

Heat is spread across multiple components, reducing hot spots.

243. How do they impact transient response?

Improved response due to faster reaction to load changes with multiple phases.

Controls/Stability

244. What does stability refer to in a buck?

The converter's ability to maintain a steady output voltage without oscillation.

245. What happens when a buck is stable/unstable?

Stable: Smooth regulation.

Unstable: Oscillation, poor regulation.

Load transients: Can cause overshoot/undershoot.

246. What type of system is the output LC filter?

A second-order system (L and C form a resonant circuit).

247. How do inductor/capacitor values impact stability?

Larger values improve stability but slow response.

248. Explain what happens during a load transient event.

The output voltage may spike or dip due to sudden changes in load, and the control loop must adjust.

249. How is transient response characterized?

By settling time, overshoot, and undershoot during load changes.

250. What is a feedforward capacitor? When is it needed?

A capacitor that improves phase margin and transient response, used to speed up feedback.

251. Explain different control topologies.

Voltage-mode: Regulates based on output voltage.

Current-mode: Regulates based on inductor current.

252. Voltage-mode vs current-mode control?

Voltage-mode: Simple, slower response.

Current-mode: Faster, better transient response, often used together for stability.

253. Explain subharmonic oscillation.

Unstable oscillation at fractions of the switching frequency, often seen without proper slope compensation.

254. What happens in peak current mode control <50% or >50% duty cycle?

<50%: Stable.

>50%: Can cause subharmonic oscillation, requiring slope compensation.

255. What is slope compensation and when is it needed?

A technique to prevent subharmonic oscillation in current-mode control when the duty cycle exceeds 50%.

Designing Buck Converters

256. What components are typically integrated in the IC?

Controller, drivers, and sometimes switches.

External: Inductors, capacitors.

257. How do you select a buck converter IC?

Based on input/output voltage range, current capacity, and efficiency requirements.

258. How should you spec the output inductor/capacitor?

Based on ripple current and ripple voltage specifications.

259. Why is the inductor's saturation current important?

It prevents the inductor from losing inductance at high currents, ensuring proper operation.

260. What determines min/max L/C?

Ripple and stability requirements; lower L and C lead to higher ripple but faster transient response.

261. How does output L/C impact stability?

Larger L/C improves stability but slows transient response.

262. How is the output voltage/current sensed?

Using a resistor divider for voltage, current-sense resistor for current.

263. Design a buck converter.

Select a buck IC, size inductor and capacitors based on voltage/current, design feedback loop.

264. How to switch with higher/lower duty cycle?

Adjust feedback voltage using a resistor network to change duty cycle.

Debugging, Validating, and Characterizing Buck Converters

265. What are some failure modes of buck converters?

Overheating, component failure, instability, and ripple out of spec.

266. What can you do to debug a buck converter?

Check for short circuits, correct inductor/capacitor sizing, and analyze feedback loop.

267. How do you validate/characterize a buck converter?

Measure efficiency, ripple, transient response, and thermal performance.

268. How does a buck converter age?

Components degrade, leading to lower efficiency, higher ripple, and potential failures.

269. How does performance vary with process, voltage, and temperature (PVT)?

Efficiency drops, ripple increases, and stability can degrade with temperature and voltage variations.

Boost Converters

270. What is a boost converter?

A converter that increases voltage from input to output.

271. How does it boost voltage?

By storing energy in the inductor and releasing it at a higher voltage.

272. Draw a boost converter circuit.

Uses a switch, diode, inductor, and capacitor.

273. What are some applications?

Battery-powered devices, where higher voltage is needed.

274. Why are they less efficient than buck converters?

More switching losses and higher inductor current.

275. What is the purpose of the diode/inductor/capacitor?

Diode: Blocks reverse current.

Inductor: Stores energy.

Capacitor: Smooths output voltage.

276. What happens to the inductor flyback voltage?

It creates the boosted voltage, critical for proper operation.

LDOs

277. What does LDO stand for?

Low Dropout Regulator.

Low voltage difference between input and output.

278. How does an LDO work?

It uses a pass transistor to regulate voltage with minimal dropout.

279. LDO as a passive component?

Can be modeled as a resistor that adjusts to maintain output voltage.

280. LDO tradeoffs vs step-down regulators?

Lower efficiency, but simpler and less noise.

281. What additional components are needed?

Input/output capacitors for stability.

282. What happens to any power losses?

Dissipated as heat.

283. LDO efficiency analysis?

Efficiency depends on the ratio of output/input voltage and output current.

Charge Pumps

284. How does a charge pump work?

It uses capacitors to transfer charge and generate a higher voltage or negative voltage.

285. How does a charge pump boost voltage?

By switching capacitors between charge and discharge states in different configurations.

286. Draw a circuit for a charge pump.

Basic circuit includes capacitors, switches, and diodes or transistors.

287. What are they used for?

Used in low-power applications to generate higher or negative voltages, like in LCD displays.

288. Charge pump efficiency analysis.

Less efficient than inductive converters, losses come from switching and capacitor charging.

289. How do you spec a capacitor for a charge pump?

Based on required voltage, current, and switching frequency. Use low-ESR capacitors for better performance.

Batteries

290. Common battery chemistries (rechargeable vs single-use)?

Rechargeable: Lithium-ion, LiPo, NiMH.

Single-use: Alkaline, Lithium primary, Zinc-carbon.

Choose based on application and reusability needs.

291. How do you characterize a battery?

By capacity, voltage, internal resistance, and energy density.

292. Desirable battery characteristics?

High energy density, long cycle life, low self-discharge, and safety.

293. Specs to look for when buying a battery?

Capacity (mAh), voltage, discharge rate (C-rating), and form factor.

294. How do you measure battery voltage/current?

Using multimeters or specialized battery monitors.

295. What to monitor in a battery for safe operation?

Voltage, temperature, and current to avoid overcharging, overheating, or over-discharging.

296. How to charge/discharge a battery safely?

Use proper charging profiles, avoid overheating, and ensure safe cutoffs for charge/discharge.

Lithium Polymer (LiPo)

297. Common battery chemistry in rechargeable consumer electronics?

Lithium Polymer (LiPo) due to its high energy density and lightweight design.

298. What does LiPo stand for?

Lithium Polymer, meaning it uses a polymer electrolyte.

299. LiPo tradeoffs vs other batteries?

Pros: High energy density, lightweight.

Cons: Safety risks, shorter lifespan compared to some chemistries.

300. Other Lithium-based battery chemistries?

Lithium-ion: Longer lifespan, more stable.

LiFePO4: Safer, but lower energy density.

301. Can LiPos be used forever?

No, they degrade over time, with capacity loss and increased internal resistance.

302. Materials in LiPos?

Lithium-based cathode, polymer electrolyte, and carbon-based anode.

303. What is internal resistance?

Resistance inside the battery that affects performance. Increases with age, temperature, and frequency.

304. What is a battery management system (BMS)?

A system that monitors voltage, current, and temperature for safety and performance optimization.

Monitoring

305. Typical LiPo cell voltage?

Nominal: 3.7V.

Range: 3.0V (empty) to 4.2V (fully charged).

306. How does temperature impact LiPo performance?

Cold: Reduces performance.

Heat: Increases risk of failure and shortens lifespan.

307. What to monitor in a LiPo circuit?

Voltage, temperature, current, and state of charge (SoC).

308. Determining state of charge (SoC)?

Using voltage or coulomb counting, but accuracy is difficult due to non-linear discharge.

309. Why is SoC accuracy important?

Ensures proper battery management, avoiding over-discharge or overcharge.

310. When is SoC accuracy more important?

Critical in high-power applications.

Less critical in low-power or intermittent use.

311. How is capacity measured?

By measuring the total charge the battery can deliver.

312. How to determine the health of a LiPo?

Check capacity loss, voltage sag, and internal resistance over time.

313. Why are charging/monitoring ICs important?

They control safe charging, monitor battery health, and prevent damage.

314. What is cell balancing?

Ensuring all cells in a battery pack have the same voltage. Important for longer life and safety.

Charging/Discharging

315. What is a LiPo's C rating?

It indicates the maximum safe charge/discharge rate relative to capacity.

316. Explain the LiPo discharge and charge curve.

Discharge: Flat at nominal voltage, then drops sharply.

Charge: Rises quickly, then levels off at the end.

317. When do LiPos charge fastest?

During the constant-current phase, before reaching 4.2V.

318. How fast can you safely charge a LiPo?

Typically at 1C (equal to its capacity), but check manufacturer's guidelines.

319. What happens if you overcharge the battery?

Can cause overheating, fire, or explosion.

320. How to charge a LiPo?

Use a LiPo charger with constant current then constant voltage (CC-CV) method.

321. What are constant-current and constant-voltage procedures?

Constant current: First phase, high current.

Constant voltage: Second phase, holds voltage while reducing current.

322. What to monitor when charging a LiPo?

Voltage, current, and temperature to prevent overcharging and overheating.

323. Design a circuit to charge/discharge a LiPo.

Use a LiPo charger IC with current sensing, temperature monitoring, and voltage cutoffs.

324. What is storage charging?

Charging to a safe middle voltage (~3.8V) for long-term storage to prevent damage.

325. Why are charging/monitoring ICs important?

They ensure safe operation and optimize battery lifespan.

326. What is balance charging?

Ensures all cells in a multi-cell battery pack are charged to the same voltage. Important in large packs.

Safety

327. Dangers of Lithium-based batteries?

Risk of fire or explosion from overcharge, overheating, or puncturing. Prevented by BMS and safe handling.

328. Why properly handle and dispose of LiPos?

Toxic materials can leak and cause fires. Always follow safe disposal methods.

329. Why is it dangerous to puncture LiPos?

Puncturing can cause internal short circuiting, leading to fire or explosion.

330. What happens if you overdischarge/overcharge a LiPo?

Overdischarge: Reduces capacity and can damage the battery.

Overcharge: Increases risk of fire or explosion.

331. What happens when you short-circuit a LiPo?

Can cause rapid heating, fire, or battery damage.

332. Why is heat a big concern?

Excess heat leads to battery degradation and increases the risk of fire. Proper thermal management is crucial

Inductive Loads

Mostly on DC inductive loads like brushed DC motors and solenoids. I've only been asked a tiny bit of AC inductive loads in the context of brushless motors. Most of these questions arise because of my personal projects with motors and solenoids, only a few roles I've interviewed for actually use motors/solenoids. I was also asked most of these questions during my freshman year before I started taking interview notes and before I had any internship or school experience, so these questions are mostly from memory. Most of these questions will be for motors (since I've been asked mostly about them), but there's a good amount of overlap with solenoids. Motors/Solenoids

333. What is a motor/solenoid? What can they be used for?

A motor converts electrical energy into mechanical motion (rotational).

A solenoid converts electrical energy into linear motion.

Used in robotics, automobiles, and electromechanical systems.

334. How does a motor/solenoid work?

Both use electromagnetic induction. Current through a coil creates a magnetic field, causing movement in a rotor (motor) or core (solenoid).

335. Why are coils often used?

Coils create a strong magnetic field when current flows through them, which is needed for generating motion in motors/solenoids.

Characterization/Behavior

336. When does a motor/solenoid draw the most current?

At startup or when it's stalled.

337. When does a motor/solenoid draw the least current?

When running at steady-state without load or under no-load conditions.

338. What happens to a motor/solenoid at steady-state under minimum and maximum current draw?

Minimum current: The motor/solenoid operates efficiently, with little force needed.

Maximum current: There's more load or resistance, causing the motor to work harder.

339. What type of loads are motors/solenoids?

Typically inductive loads due to their coils.

340. What is flyback voltage?

The voltage spike generated when current through an inductive load like a motor or solenoid is suddenly interrupted.

341. What is back EMF?

The voltage generated by a motor that opposes the current driving it, occurring when the motor spins.

342. Will I see 12A of current being drawn from a 12V supply into a 1-ohm coil motor?

No, because of the inductance. Inrush current may be high initially, but back EMF and inductance will limit the steady-state current.

343. What is inrush current?

The initial surge of current when the motor or solenoid is powered on, before the inductance stabilizes the flow.

344. When do motors/solenoids draw the most current?

At startup or during a stalled condition.

345. Why do motors/solenoids still experience inrush current despite inductors opposing instantaneous current changes?

Inductance opposes current change, but inrush occurs because there's no back EMF at startup, allowing high current until the motor speeds up.

Brushed DC Motors

346. Difference between motor and generator?

Motor: Converts electrical energy into mechanical energy.

Generator: Converts mechanical energy into electrical energy.

347. Difference between motor and solenoid?

Motor: Produces rotational motion.

Solenoid: Produces linear motion.

348. Parts of a motor (rotating/static)?

Rotor (armature) rotates.

Stator and commutator remain static.

349. Types of motors and their uses?

DC motor: Small electronics, toys.

Stepper motor: Precision control in CNC machines.

BLDC: Drones, e-bikes.

350. What are linear motors?

Motors that produce linear motion instead of rotational, used in maglev trains and actuators.

351. Motor for phone vibration?

Eccentric rotating mass (ERM) motor or linear resonant actuators (LRA).

352. Torque vs RPM curve implications?

Torque decreases as RPM increases. High torque is available at low RPMs; as RPM increases, torque drops.

353. Electrical parameter for high torque?

Current drives high torque. High RPM corresponds to high voltage.

354. How do you spec a motor?

Based on voltage, current, RPM, and torque requirements.

355. Why are motors noisy? How to reduce noise?

Noise comes from vibration, bearing friction, and brush arcing. Reduce it by using brushless motors or adding damping.

356. What is arcing?

Sparks formed when the brushes make/break contact with the commutator.

Brushes

357. What are brushes?

Conductive components that transfer current between the rotating rotor and stationary stator.

358. What are brushes made out of?

Commonly made of carbon or graphite.

359. Concerns about brushes?

Wear out over time and require maintenance.

360. Why do brushes wear out?

Due to friction and electrical arcing at the commutator.

361. Replace brushes or motor?

Replace brushes if they're worn but the motor is still functional. Replace the motor if it has other failures.

362. How do brushes maintain contact with the commutator?

They are spring-loaded, ensuring constant contact with the commutator surface.

Brushless DC (BLDC) Motors

363. How do BLDC motors work?

BLDC motors use electronic commutation instead of brushes, with a permanent magnet rotor and a stator with wound coils.

364. How do you control BLDC motors?

Using an electronic speed controller (ESC) that switches the stator coils based on rotor position.

365. Control a BLDC motor from a microcontroller?

Use an ESC or motor driver, and control the motor with PWM signals from the microcontroller.

366. Advantages of brushless vs brushed motors?

Higher efficiency, longer lifespan, quieter, and less maintenance.

367. What are motor poles?

Poles refer to the number of magnetic poles in the rotor. More poles provide smoother operation but may lower top speed.

368. How to spec a BLDC motor?

Based on voltage, RPM per volt (KV), and current rating.

Electronic Speed Controllers (ESCs)

369. What is an ESC?

An ESC is a circuit that controls the speed and direction of BLDC motors by switching power to the motor’s coils.

370. Why are ESCs required for brushless motors?

To electronically commutate the motor coils, since brushless motors don’t have physical commutators.

371. Difference between ESC and single-phase motor driver?

ESCs control multiple phases for BLDC motors, while single-phase drivers are simpler, used for DC motors.

372. Components of an ESC?

Microcontroller, MOSFETs, gate drivers, current sensing, and PWM controller.

373. Interface with ESCs?

Communicate using PWM signals from a microcontroller or RC receiver.

374. Design an ESC.

Includes a microcontroller for control, MOSFET drivers for switching, and feedback loops for speed regulation.

375. How to spec an ESC?

Match to the motor's voltage, current, and power requirements.

Stepper Motors

376. What is a stepper motor?

A motor that moves in discrete steps, controlled by electrical pulses.

377. How does it work?

Energizing coils in sequence causes the rotor to move step by step.

378. Applications?

CNC machines, 3D printers, and robotics requiring precise positioning.

379. Advantages compared to other motors?

Precise control of position without needing feedback sensors.

380. Control a stepper motor from a microcontroller?

Use a stepper motor driver to send step pulses and direction signals from the microcontroller.

381. How to spec a stepper motor?

Based on step angle, torque, current, and voltage.

382. How do stepper motors hold a position?

They can hold position due to their magnetic fields and the detent torque between steps.

Servo Motors

383. What is a servo motor?

A servo motor is a type of motor that provides precise control of angular or linear position, velocity, and acceleration. It uses a feedback sensor (typically a potentiometer or encoder) to maintain accurate positioning.

384. How does it work?

A servo motor operates via a closed-loop control system. A control signal (typically PWM) determines the desired position, and feedback from the motor's sensor continuously adjusts its movement to match the target position. An internal controller adjusts the motor's power to maintain the specified position or motion.

385. What are its applications?

Servo motors are commonly used in robotics, automated machinery, RC vehicles, CNC machines, and precise positioning systems where high accuracy is required.

386. What are its advantages compared to other types of motors?

Precise control of position and speed

Closed-loop feedback ensures accurate positioning

High torque at low speeds

Ideal for applications requiring repeated motion patterns

387. How do you control a servo motor from a microcontroller? What other components do you need?

A servo motor is controlled using PWM signals from a microcontroller. The width of the PWM signal determines the position of the servo. You need a power supply, PWM output pin, and possibly driver circuitry for more powerful servos.

388. How do you spec a servo motor?

To spec a servo motor, consider:

Torque rating: Based on the load it needs to move

Speed: How fast it needs to reach the desired position

Voltage and current: Power requirements

Angular range: The maximum rotation angle (e.g., 180° or continuous)

389. How are servo motors so precise in their movement?

Servo motors use feedback loops with a sensor (e.g., potentiometer or encoder) that constantly provides information about the current position. This feedback allows the controller to make real-time adjustments, ensuring precise movement to the desired position.

Safety and Monitoring

390. What are some failure modes of motors/solenoids? How can they be avoided?

Overheating: Use thermal cutoffs and proper cooling

Overcurrent: Use fuses or current-limiting circuits

Mechanical wear: Regular maintenance, lubrication, and inspection

Insulation failure: Use high-quality insulation material and avoid overloading

391. What are some dangers when using motors/solenoids?

Electrical hazards: Short circuits, improper grounding

Mechanical hazards: Moving parts can cause injury, use guards

Fire risk: Overheating or arcing can ignite materials

392. What properties do you want to measure when using a motor/solenoid?

Current draw

Temperature

Position (for motors)

Speed (for motors)

Force or torque

393. How can these properties be monitored? What sensors/components can you use?

Current sensor for current draw

Thermocouple or RTD for temperature

Rotary encoders or potentiometers for position

Hall effect sensors for speed

Load cells for force/torque

394. How would you alter the behavior of the motor/solenoid based on the observed data?

Adjust current to prevent overheating

Modify PWM duty cycle to change speed/position

Implement feedback control loops (e.g., PID) for precise control

395. How do you determine how much current a motor/solenoid is drawing?

You can measure current using a current sensor (e.g., a shunt resistor or Hall effect sensor) placed in series with the motor/solenoid.

396. How can you detect the temperature of a motor/solenoid?

Use a thermocouple, thermistor, or RTD to measure the motor’s surface temperature or internal temperature near the windings.

397. Which temperatures do you want to/can you measure?

Measure winding temperature to avoid overheating

Surface temperature of the motor/solenoid housing

398. What are some contactless ways to measure temperature?

Infrared (IR) sensors

Thermal cameras

These allow temperature measurement without physical contact.

399. How do motors perform as they age?

Decreased efficiency due to wear in bearings or brushes

Increased noise

Possible higher current draw and reduced torque

400. What kind of stress may a motor experience in its lifetime?

Thermal stress from heat cycling

Mechanical stress from vibrations or heavy loads

Electrical stress from voltage spikes or overcurrent

401. Do motor/solenoid coils burn out from thin wires?

Coils don't burn out if designed properly with appropriate current ratings and cooling mechanisms. Using multiple strands of thin wire (Litz wire) can increase current capacity without overheating.

Speed/Position

402. When/why would knowing the speed and/or position of a motor/solenoid be useful?

For precise control in robotics, CNC machines, or automated systems

Safety: Prevent overspeed or incorrect positioning

403. How can you measure the RPM of a motor?

Use a tachometer or Hall effect sensor

Rotary encoder can also track RPM

404. How can you determine the position of a motor?

Use a rotary encoder or potentiometer to measure shaft position

405. What is a rotary encoder? How can it be used to measure motor speed?

A rotary encoder detects the angular position of a rotating shaft. By counting the number of pulses per second, you can measure the motor's speed (RPM).

406. How can you measure the speed of a solenoid?

Use a position sensor (e.g., Hall effect or LVDT) to track motion over time.

407. How can you measure the position of a solenoid?

Use a linear potentiometer or LVDT (Linear Variable Differential Transformer).

408. How can you use BEMF to measure motor/solenoid speed?

By measuring the back EMF generated when the motor is running, you can estimate its speed, as BEMF is proportional to motor velocity.

Single-Phase Inductive Drives

PCB design/layout on inductive drives is in the PCBs section.

409. How do you drive an inductive load?

Use a PWM-controlled driver circuit with diodes to handle flyback voltage, and include current-limiting features.

410. What if you need motor braking?

Use a brake resistor or apply reverse voltage to stop the motor.

411. What if you need to drive the motor in both directions?

Use an H-bridge to switch the polarity and allow bidirectional control.

412. What are the different kinds of motor braking?

Dynamic braking: Applying resistance to dissipate energy

Regenerative braking: Converts kinetic energy back into electrical energy

Plugging: Reversing polarity to rapidly stop the motor

413. What sort of protection circuitry do you need?

Flyback diodes

Fuses or circuit breakers

Thermal cutoffs

414. Tradeoffs between different drive circuits: EMR, SSR, MOSFET, BJT, IGBT?

EMR: Reliable but slow and noisy

SSR: Quiet but expensive

MOSFET: Efficient, high-speed switching

BJT: Less efficient, slower switching

IGBT: Suitable for high-voltage applications

415. Optimizations for solenoid drives?

Use higher voltages with current limiting to increase plunger velocity and cycle rate.

416. Specifying a driver IC?

Consider voltage and current ratings, switching speed, and thermal management.

417. Specifying transistors for a motor/solenoid drive?

Look at current rating, voltage rating, Rds(on) (for MOSFETs), and switching frequency.

418. What limits the switching frequency?

Gate capacitance of the transistor

Parasitic inductance

419. How do you increase the switching frequency?

Use transistors with lower gate capacitance

Minimize parasitic inductance through proper PCB layout.

Half/H-Bridge

There's a good amount of overlap with the Power Bridge subsection of Buck Converters as well, since the power bridge of a buck is often just a half bridge.

420. Draw an H-Bridge Circuit:

An H-Bridge is a way to control a motor so it can turn in both directions (forward and reverse). It uses four switches (usually transistors) arranged in an "H" shape. By turning on different pairs of switches, you can make the motor turn one way or the other.

421. Draw a Half-Bridge Circuit:

A Half-Bridge is a simpler version with only two switches. It can control current in just one direction, so the motor only spins in one direction (but can be turned off).

422. How Does a Half/H-Bridge Work?

Half-Bridge: Controls current in one direction. Simple but can't reverse the motor.

H-Bridge: Controls current in both directions, so it can make the motor spin forward or backward.

423. What is the Difference Between an H-Bridge and Half-Bridge? What Are the Tradeoffs?

H-Bridge: Lets you control the motor in both directions but requires more components.

Half-Bridge: Only controls one direction, simpler but less flexible.

424. How Can You Brake a Motor Using a Half-Bridge? What About an H-Bridge? Which Can Brake the Motor Faster? How Does This Stress the Motor?

Half-Bridge braking: You can stop the motor by turning on both switches, which shorts the motor, slowing it down.

H-Bridge braking: You can brake even faster by shorting both sides of the motor.

Faster braking creates more heat and wear on the motor.

425. When Can You Use an Asymmetrical Half-Bridge? When Do You Need a Symmetrical H-Bridge?

Asymmetrical Half-Bridge: Used when you only need to spin the motor in one direction.

Symmetrical H-Bridge: Needed when you want the motor to go both forward and backward.

426. What Are the Advantages of Using a High-Side NFET? What Additional Challenges Are Presented?

Advantages: NFETs waste less power and can handle more current, so they are more efficient.

Challenges: To turn on a high-side NFET, you need to provide a higher voltage than usual, which requires extra circuitry.

427. Design a Circuit to Properly Drive a High-Side NFET. Why Is It Needed?

To make a high-side NFET work, you need a special circuit (called a gate driver) that gives a higher voltage to the NFET's gate, so it can turn on properly. This is important when you're controlling high-power loads like motors.

428. What Happens if You Turn Both the PFET and NFET on the Same Side at the Same Time? How Do You Prevent This?

If both switches (PFET and NFET) on the same side of the circuit are on at the same time, it creates a short circuit, which can damage your components. To prevent this, you add a small delay between turning them on and off (called dead time) so they never overlap.

Control from Embedded Device

Often times you want to control a big motor/solenoid from a small embedded device. This section focuses on some of the more power electronics side of things while the Embedded Systems section will focus a bit more on the embedded side of things.

429. Design a circuit to drive an inductive load from a microcontroller GPIO:

To drive an inductive load (like a motor or solenoid) from a microcontroller, you'll typically need a transistor (MOSFET) that acts as a switch. The microcontroller GPIO controls the MOSFET gate, allowing current to flow through the load when needed. A flyback diode is added across the load to protect the circuit from voltage spikes generated when switching off the inductive load.

430. Why would a pulldown resistor be necessary?

A pulldown resistor ensures that the MOSFET gate stays at a known voltage (ground) when the GPIO pin is not driving it. This prevents the MOSFET from turning on unexpectedly due to floating voltages.

431. Why would you want a series resistor between the GPIO and MOSFET gate? What is parasitic oscillation in the context and where can it arise from? What does the series resistor do?

The series resistor limits the current charging/discharging the MOSFET gate, slowing the switching and helping prevent parasitic oscillations, which are unwanted high-frequency signals caused by the interaction between the MOSFET gate capacitance and wiring inductance. The resistor stabilizes the circuit.

432. Why is isolation important? What are the tradeoffs?

Isolation is important to protect low-power circuits (like a microcontroller) from high-power sections (like a motor) and prevent ground loops. Tradeoffs include added complexity, cost, and reduced signal speed in some isolation methods.

433. What sort of isolations would you want?

You would want electrical isolation between the control side (microcontroller) and the power side (motors, solenoids), typically achieved through opto-isolators or transformers.

434. What is opto-isolation? What are its tradeoffs?

Opto-isolation uses light to transfer signals between two electrically isolated circuits, providing good isolation. Tradeoffs include slower signal speeds and increased cost and complexity.

435. Describe/draw/design an opto-isolation circuit:

In an opto-isolator, an LED on one side emits light when a signal is sent, and a phototransistor on the other side detects the light, thus transferring the signal while keeping both circuits electrically isolated.

436. What is galvanic isolation?

Galvanic isolation refers to complete electrical separation between two circuits, ensuring no direct current can flow between them. It prevents ground loops and protects sensitive components.

437. What are some challenges when using a high-side PFET and a much higher supply voltage to power the motor?

The challenge with using a high-side PFET is that the gate voltage must be higher than the source voltage (which is already high due to the power supply), making it hard to turn the PFET on. Special circuits (like charge pumps) are needed to achieve this.

438. Design a circuit to properly drive a high-side PFET. Why is this needed?

A typical solution involves using a gate driver to boost the voltage applied to the PFET gate, allowing it to switch properly. This is needed to control high-power loads while maintaining isolation between the control and power circuits.

Current Sensing

439. How can you measure current?

You can measure current by using a current sense resistor (voltage drop across it) or a Hall effect sensor that detects the magnetic field generated by the current.

440. Why do you want to measure current?

Measuring current is important to monitor power usage, detect faults, protect circuits from overcurrent, and improve efficiency.

441. How does inductive/magnetic current sensing work? What are its pros and cons?

Inductive sensing uses a coil or Hall effect sensor to detect the magnetic field produced by current flow. Pros: No direct connection to the current path (non-invasive). Cons: May have lower accuracy compared to direct methods like resistive sensing.

442. Describe tradeoffs between different current sensing mechanisms:

Shunt resistors: Accurate but introduce power loss.

Hall effect sensors: Non-invasive, but more expensive.

Current transformers: Suitable for AC but can't measure DC.

Sense Resistor

443. How does a sense resistor work to sense current?

A sense resistor creates a small, measurable voltage drop proportional to the current flowing through it (Ohm's Law: $V = I \times R$). This voltage is then measured to calculate current.

444. What are its pros and cons?

Pros: Simple, accurate, and inexpensive. Cons: Causes a small voltage drop, which can affect circuit efficiency.

445. How do you spec a current sense resistor? What happens if it's too big/small?

You select a resistor based on the expected current and acceptable voltage drop. Too big: Large voltage drop, reduced efficiency. Too small: Low voltage drop, harder to measure accurately.

446. Given a range of input currents and ADC input voltage range, find an appropriate sense resistor value:

You would calculate the resistor value based on the maximum current you expect and the maximum measurable voltage of the ADC. Use Ohm's law to find the resistor that fits within the ADC range.

447. What is the difference between high-side and low-side current sensing? What are the advantages/disadvantages of each? When do you need to use one vs the other?

High-side sensing measures current between the power supply and load. Advantage: Load is grounded. Disadvantage: Requires specialized components due to high voltage.

Low-side sensing measures current between the load and ground. Advantage: Easier to design. Disadvantage: Load is not directly grounded, which may cause ground disturbances.

448. How can you use drain-source measurements of a MOSFET to determine current? What are the pros and cons of this setup?

The voltage across the MOSFET's drain and source can give an estimate of current using its Rds(on) (on-resistance). Pros: No extra components. Cons: Accuracy depends on the MOSFET's temperature and Rds(on) variation.

Current Sense Amplifiers (CSA)

449. What is a current sense amplifier (CSA)?

A current sense amplifier (CSA) amplifies the small voltage drop across a sense resistor to a level that can be easily measured by an ADC or microcontroller.

450. When do you need a CSA?

You need a CSA when the voltage drop across the sense resistor is too small for direct measurement by the ADC, or when high precision is required.

451. What is the difference between a CSA and a regular opamp?

A CSA is specifically designed for accurate current measurement with built-in features like high common-mode rejection, whereas a regular opamp is more general-purpose and may lack these features.

452. How do you spec a CSA?

When selecting a CSA, consider the common-mode voltage, gain, input offset voltage, and bandwidth based on your application.

453. Given a range of input currents, a sense resistor value, and ADC input voltage range, find an appropriate CSA gain:

The CSA gain is calculated to amplify the maximum expected voltage drop (based on the sense resistor and max current) to fit within the ADC's input voltage range. Divide the ADC range by the sense resistor's voltage drop to find the gain.

Current/Voltage Protection

454. How/why do you want to protect against overcurrent and/or over/undervoltage conditions?

Protection ensures your circuit doesn't get damaged by excessive current or incorrect voltages, which can cause overheating, fire, or component failure.

455. Design a circuit to detect overcurrent and/or over/undervoltage conditions:

You can use a current sense circuit combined with a comparator to detect when current exceeds a preset value. For voltage, use a voltage divider and comparator to monitor voltage levels.

456. If an overcurrent and/or over/undervoltage condition is detected, what can you do?

You can trigger a protection circuit to shut down the power supply or reduce the load to prevent damage.

457. How/why do you need to protect a power supply from overcurrent/overvoltage?

Overcurrent or overvoltage can damage components, reduce the lifespan of the power supply, and cause safety hazards. Protection ensures safe operation.

458. What happens if you get an over/undervoltage on an embedded system?

Overvoltage can damage components or cause malfunction. Undervoltage can lead to erratic behavior or cause the system to reset.

459. How can you protect a load from pulling too much current?

You can use current-limiting circuits, fuses, or electronic circuit breakers to protect the load from excessive current.

Printed Circuit Boards (PCBs)

460. What is a PCB? Name some applications.

A PCB (Printed Circuit Board) is a board that connects electronic components using thin copper traces instead of wires. It's used in many electronic devices, like smartphones, computers, and home appliances.

461. Tradeoffs of PCBs vs solderless breadboards vs soldered breadboards vs wires vs ICs?

PCBs: Permanent, reliable, compact, but costly to design.

Solderless Breadboards: Easy to experiment with, but not reliable for final products.

Soldered Breadboards: More durable but not as neat as a PCB.

Wires: Flexible but messy and prone to errors.

ICs: Integrated circuits, which combine many components into one chip, making circuits more compact but harder to modify.

462. Difference Between a PCB and IC?

PCB: A board that connects components.

IC: A chip that contains miniaturized circuits inside it.

463. Parasitic Elements on a PCB?

Parasitic elements are unwanted effects like stray capacitance or resistance that affect how a PCB performs, especially at high speeds.

464. Different Types of PCBs? Tradeoffs and Applications?

Single-layer PCB: Simple, cheap, used in basic electronics.

Multi-layer PCB: Complex, more layers, used in advanced devices like smartphones.

Flexible PCBs: Can bend, used in tight spaces but more expensive.

465. What is a Flex PCB? Tradeoffs and Applications?

A flex PCB can bend and fold, making it useful in devices like wearables or foldable screens. However, they're more expensive and harder to manufacture.

Layers and Materials

466. What are the layers of a PCB? What are they made of?

PCBs have layers like:

Copper layer (for conducting signals)

Insulation layer (FR4 or other materials)

Soldermask (protective coating)

Silkscreen (for text labels).

467. What material is used for conduction layers?

Copper is typically used because it's a great conductor.

468. Tradeoffs of using gold vs copper for conduction layers?

Gold: More resistant to corrosion but expensive.

Copper: Cheaper and commonly used, but can oxidize if unprotected.

469. Materials for insulation? Tradeoffs?

FR4: Strong and cheap, but not flexible.

Polyimide: Used in flex PCBs, flexible but more costly.

470. What does "FR4" mean?

FR4 is a common insulating material made of fiberglass and epoxy, used in most PCBs.

471. Why are PCBs often green? What other colors could they be?

Green is the traditional color for the soldermask, but other colors like red, blue, and black are also available.

472. What is the layer with text called?

The silkscreen layer is where text or labels are printed on a PCB.

473. What is soldermask?

The soldermask is the protective layer that covers the copper traces, preventing short circuits and corrosion.

Stackup

474. What are the layers of a PCB? What are they made of?

PCBs typically have copper layers for conduction, insulation layers (FR4), a soldermask, and silkscreen for labeling.

475. How to choose how many layers for a PCB?

More layers are needed for complex designs (like in computers) to separate signals, power, and ground.

476. How to choose board stackup?

It depends on the complexity of the circuit, speed of signals, and power distribution needs.

477. What should be routed on each layer?

Top/bottom layers: Components and simple connections.

Inner layers: Power, ground, or high-speed signals.

478. Which layer(s) should be the ground layer(s)?

One of the inner layers is typically a ground plane, for a stable reference.

479. Which layers should high-speed signals be routed?

Inner layers are best for high-speed signals to minimize interference.

480. Which layers should power planes be on?

Usually on a dedicated inner layer, separate from the ground.

481. What components require a higher layer count?

Complex chips, like microprocessors or FPGAs, often require more layers for routing signals, power, and ground.

Design/Layout/Routing

482. General rule of thumb for board routing?

Keep traces short and wide for power, avoid crossing signals, and place decoupling capacitors close to components.

483. How do you decide which subsystems go where on a PCB?

Group components logically (e.g., power components together) and keep critical circuits away from noisy ones.

Grounding

484. What is grounding on a PCB? Why is it important?

Grounding connects all components to a common reference point, preventing noise and improving signal integrity.

485. Basic grounding layout techniques?

Use a ground plane, avoid ground loops, and keep ground paths short and direct.

486. What is a ground plane? Pros/cons?

A ground plane is a solid layer of copper that reduces noise. It's great for signal quality but takes up space.

487. What is ground stitching?

Connecting multiple ground planes with vias to ensure they are at the same potential.

488. Ground plane vs ground traces?

A ground plane provides better performance by reducing noise, but ground traces save space.

489. Other functionalities of ground planes?

They help with shielding signals and heat dissipation.

490. Why can you use a ground plane to shield signals?

The ground plane acts as a barrier to prevent interference from other signals.

491. Which layers typically have ground planes and why?

Inner layers are used for ground planes because they are shielded by other layers, reducing noise.

492. Why are ground probe points important?

They provide easy access for testing and debugging.

493. When/why should you separate your ground reference potentials?

Separate ground planes for analog and digital circuits to reduce interference.

494. What are some ground layout schemes?

Star grounding or having a single ground point is common to avoid loops.

495. What is star grounding?

All ground connections come to a single point to avoid ground loops.

496. Combining analog, digital, and power grounds vs separating them?

Combining is simpler, but separating them can reduce interference in sensitive circuits.

497. What happens if your ground plane potential differs across your board?

It can cause noise, signal distortion, or improper functioning of circuits.

498. What causes ground plane potential variation?

Long or thin ground paths, poor connections, or high current can cause differences in potential.

499. How do you avoid too much variation?

Use solid ground planes, and keep paths short and thick.

500. How can you design your system to be more resilient against ground disturbances?

Use proper grounding techniques, shield critical signals, and add bypass capacitors for noise reduction.

Decoupling/Bypass Capacitors

501. What are decoupling capacitors? When are they needed?

Decoupling capacitors smooth out voltage supply noise. They're needed next to power pins of chips to ensure clean power.

502. Common values for decoupling capacitors?

Values like 0.1 µF are common for general use, while others depend on the frequency.

503. Rules of thumb when placing and routing decoupling capacitors?

Place them as close as possible to the chip's power pins and use short, wide traces.

504. Why place decoupling capacitors close to the source package?

The closer they are, the more effective they are at filtering out noise.

505. What signals do you often want to decouple? Why?

Power signals, because fluctuations can cause unstable operation in digital circuits.

506. Why are there many decoupling capacitors on a signal?

Different capacitors handle different frequencies of noise. Multiple capacitors provide better overall filtering.

507. How do you lay out multiple decoupling capacitors?

Place the smaller capacitors closest to the chip, followed by the larger ones.

508. What happens if decoupling capacitors are placed too far from the package?

They become less effective at filtering noise, leading to potential performance issues.

509. How does parasitic inductance impact decoupling capacitor performance?

High parasitic inductance reduces the effectiveness of capacitors at high frequencies.

510. Sources of parasitic inductance that may decrease decoupling capacitor effectiveness?

Long traces and vias add parasitic inductance.

511. How can packaging impact effectiveness?

Capacitors with smaller packages generally perform better because they reduce parasitic inductance.

512. What is a capacitor's self-resonant frequency? Why is it important?

The frequency at which a capacitor stops being capacitive and becomes inductive. It's important to choose capacitors that resonate at the right frequencies.

513. When should you not use decoupling capacitors?

In circuits where noise filtering isn't critical, such as low-speed or analog circuits.

High-Speed Signals

Diff pairs and high-speed design from board layout/design perspective, as opposed to signals or SI perspective. Diff pairs are often used for high-speed signals, so there will be a good amount of overlap with the Transmission Lines (TL) section, especially microstrip TLs.

514. What are some rules of thumb when routing high-speed signals?

Keep traces short and straight to minimize delays and interference.

Avoid sharp corners; use 45° angles instead.

Ensure consistent impedance by controlling trace width and spacing.

Isolate high-speed signals from noisy power lines and sensitive analog signals.

515. Should you put vias through high-speed traces?

Avoid using vias for high-speed traces because they introduce parasitic inductance and capacitance, which can distort the signal. If vias are necessary, minimize their use.

516. What shape should these high-speed traces be? Why?

High-speed traces should be straight and smooth, with 45° turns instead of 90° angles to prevent signal reflections and minimize impedance changes.

517. What is serpentining? Why is it important for synchronization?

Serpentining is the practice of meandering a signal trace to match the length of another trace, ensuring that both signals reach their destination at the same time. This is important for synchronizing differential signals.

Differential Pairs

518. What is a differential pair? What are some tradeoffs?

A differential pair consists of two traces carrying equal and opposite signals, used to reduce noise and interference. Tradeoffs include needing precise trace length matching and spacing, which can complicate layout.

519. How should it be routed? What are some layout guidelines that differ from single-ended high-speed signals?

Differential pairs should be routed:

Close together with consistent spacing to maintain the same impedance.

Traces should have equal lengths to avoid skew.

Minimize vias to avoid signal degradation.

520. Why is it important that signals for both lines of the diff pair arrive at the source at the same time? What happens if they don't? How can you ensure that they do?

If signals don't arrive at the same time, it causes skew, leading to signal distortion and data errors. To ensure simultaneous arrival, use equal trace lengths (length matching) and serpentining if necessary.

521. What is serpentining? Why is it important for diff pairs?

Serpentining in differential pairs ensures that both traces are of equal length, which is critical for signal integrity. Uneven trace lengths would cause phase mismatch and degrade signal quality.

Transmission Line Effects

Transmission line (TL) effects on a PCB/microstrip transmission line, also lots of overlap with TL section.

522. Describe transmission line components on a PCB.

A transmission line consists of a signal trace and a reference plane (ground). It's modeled as a combination of resistance, capacitance, inductance, and conductance.

523. What are some common characteristic impedance values?

Common values are 50 ohms (single-ended) and 100 ohms (differential pairs).

524. How can you change the characteristic impedance of a trace?

You can adjust the trace's width, the distance to the ground plane, and the dielectric material to control impedance.

525. How does a trace's impedance change with respect to frequency?

At higher frequencies, a trace's impedance can change due to skin effect, where current travels more on the surface of the conductor, increasing resistance.

526. Describe some sources of reflection on a PCB. How can these reflections be minimized?

Reflections occur when impedance is mismatched (e.g., at connectors or vias). Minimize them by using matched impedance and termination resistors.

527. How do any interfaces in the transmission path impact signal quality and reflections?

Connectors, vias, or improper terminations can introduce impedance changes, causing signal reflections and distortion. Proper termination and impedance matching help reduce these effects.

528. If the substrate layer is twice as thick, how does this impact characteristic impedance? How could you fix it?

A thicker substrate increases the characteristic impedance. You could fix it by increasing the trace width to bring impedance back to the desired value.

Board-Level Termination

529. If external termination components are used, where should they be placed?

External termination components should be placed near the load (sink) to properly absorb signals and prevent reflections.

530. How do you route external termination components?

Route them as close to the signal trace and load as possible, keeping the traces short to avoid adding parasitic effects.

531. When would you need external termination components?

External termination is needed when dealing with high-speed signals to minimize reflections and ensure signal integrity.

532. Describe some termination networks.

Series termination: A resistor is placed in series with the signal near the source to match the impedance.

Parallel termination: A resistor is placed between the signal and ground near the load.

533. Difference between series and parallel termination?

Series termination reduces reflections but adds a slight delay.

Parallel termination improves signal quality but consumes more power.

High-Current Layout/Routing

534. What are some rules of thumb when laying out and routing high-current connections?

Use thicker traces to handle the current.

Keep traces short to reduce voltage drop.

Use multiple vias to connect layers for better current handling.

535. What is a power plane? Why are they preferred over traces?

A power plane is a large copper area that distributes power across the board. It’s preferred over traces because it can carry more current with less resistance and heat.

536. Are long shapes/traces preferred over short ones? Why?

Short traces are preferred because they reduce resistance and voltage drop.

537. Are skinny shapes/traces preferred over thick ones? Why?

Thicker traces are better for high current because they reduce resistance and heat generation.

538. What is a thermal relief pad?

A thermal relief pad is a special pad design that connects a pad to a plane using thin traces to make it easier to solder without heating the entire plane.

539. What is via stitching? When can it be used?

Via stitching is the practice of placing multiple vias to connect copper planes. It is used to improve electrical conductivity and thermal performance.

540. What can you do at the board level to reduce heating of a component/IC?

Use heat sinks, thermal vias, or place the component near a power plane for better heat dissipation.

541. When should heat sinks be used?

Heat sinks are used when a component generates significant heat, and natural airflow or copper planes aren't enough to keep it cool.

Power Supply Layout/Routing

542. What are some considerations when deciding where to place the power supply?

Place the power supply close to the load to minimize voltage drops and keep high-current paths short.

543. Why do you want to place your power supply close to the load?

To reduce the resistance and inductance of the traces, which can cause voltage drops or noise.

544. What happens if your load tries to draw too much current but there's lots of trace distance between your load and power supply?

You may experience voltage drops and signal integrity issues due to the resistance and inductance in the long traces.

Component-Level Layout/Routing

545. What are some considerations when laying out power supply components?

Place critical components (e.g., capacitors, inductors) close together and minimize trace lengths for high-current paths to reduce noise and voltage drops.

546. Important nodes of switch-mode power supplies? Why?

The switching node and input/output nodes are critical because they handle high currents and are prone to noise, so they need careful routing.

547. Noisy nodes of a power supply? How to route them?

The switching node is noisy. Keep it away from sensitive circuits and use short, direct traces to minimize noise.

548. How should you route the input capacitor?

Place it as close as possible to the power input pin to reduce noise and voltage drop.

549. What if the input capacitor is placed too far?

If the capacitor is too far, it won't filter noise effectively, and you may experience instability or ripple.

550. How should you route the switch node of a buck converter?

Keep the switch node traces short to reduce noise and prevent EMI.

551. How should you route the inductor of a buck converter?

Route the inductor close to the switch node and output capacitor, using wide traces to handle high currents.

552. How should you route the sense/feedback network of a buck converter?

Keep sense/feedback traces away from noisy components and shield them with ground planes if possible.

553. How do you route decoupling and bulk capacitors?
Place them close to the IC pins, using short traces to reduce inductance and improve filtering efficiency.

Sense Resistor Layout/Routing

554. How do you route current sense resistors?

When routing current sense resistors, you should aim for minimal inductance and resistance in the traces. Connect the sense resistor in series with the load to measure current. Use a dedicated pair of traces for the voltage drop across the resistor, and ensure these traces are as short and direct as possible to avoid errors introduced by trace resistance and inductance.

555. Why is it important that the traces be short?

Short traces minimize the impact of parasitic inductance and resistance, which can distort the voltage measurement across the sense resistor and lead to inaccurate current readings. Longer traces can pick up noise and can affect the precision of the measurement.

556. What if your sense resistor is far from your measurement circuitry?

If the sense resistor is far from the measurement circuitry, you should use Kelvin (or 4-wire) connections to minimize the impact of trace resistance and inductance. This means you will have separate pairs of traces: one pair for carrying the current and another pair for measuring the voltage drop across the resistor.

557. What is a Kelvin connection for a sense resistor? What are the tradeoffs?

A Kelvin connection, or 4-wire measurement, uses two wires to carry the current through the resistor and two additional wires to measure the voltage drop across the resistor. This setup eliminates errors due to the resistance of the measurement wires and trace resistance. The tradeoff is that it requires more board space and additional wiring, which can increase complexity and cost.

Noise, Interference, and Coupling

558. Why is noise a consideration at the board-level?

Noise at the board level can interfere with signal integrity, leading to data errors, reduced performance, and reliability issues. Noise can come from various sources, including power supply fluctuations, switching circuits, and external electromagnetic interference (EMI).

559. What is electromagnetic interference (EMI)?

EMI is the disturbance caused by electromagnetic fields that can disrupt the operation of electronic devices and systems. It can be generated by various sources such as power supplies, high-frequency circuits, and other electronic equipment.

560. What are some sources of EMI on a PCB?

Sources of EMI on a PCB include switching power supplies, high-speed digital signals, oscillators, and clock lines. These components can emit electromagnetic fields that interfere with nearby signals and circuits.

561. Why do I want to avoid radiated EMI?

Radiated EMI can affect the performance and reliability of electronic circuits by inducing noise in nearby traces or components. It can also lead to regulatory compliance issues, as many standards require electronics to operate within specified EMI limits.

562. What are some board-level techniques to minimize radiated EMI?

Techniques to minimize radiated EMI include proper grounding, shielding critical traces with ground planes, using low-impedance connections, placing sensitive components away from high-frequency signals, and employing EMI shielding materials.

563. What about external sources of EMI? How can they impact a circuit?

External EMI sources, such as radio transmitters or power lines, can induce unwanted signals into a circuit, leading to potential malfunctions or degraded performance. To mitigate these impacts, circuits can be shielded, and proper filtering and grounding practices can be implemented.

564. What are some board-level techniques to protect against EMI?

Techniques to protect against EMI include using shielding enclosures, implementing filtering components like capacitors and inductors, designing PCB layouts with careful trace routing, and incorporating proper grounding techniques.

565. How can signals couple together? How can this be prevented?

Signals can couple together through capacitive, inductive, or resistive coupling. Capacitive coupling occurs when signals share a common electric field; inductive coupling happens when signals share a magnetic field; resistive coupling involves shared ground paths. To prevent coupling, maintain adequate spacing between high-speed and sensitive signals, use shielding, and employ differential signaling where appropriate.

566. What are some board-level techniques to minimize coupling?

Techniques to minimize coupling include using ground planes, separating high-frequency and sensitive analog signals, using twisted pair or differential signals for high-speed lines, and implementing proper shielding and trace routing practices.

567. If I have two traces on adjacent layers, one a clock and another a sensitive analog signal, how should I route these signals?

Route high-speed signals like clocks away from sensitive analog signals. Place them on separate layers with ground planes in between to provide isolation. If traces must cross, ensure they do so perpendicularly to minimize coupling.

568. What is desense? Why should it be avoided? What are some board-level techniques to avoid it?

Desense refers to the reduction of signal quality due to interference from other signals or noise sources. It should be avoided because it can lead to signal degradation and malfunction. Techniques to avoid desense include careful PCB layout, shielding, adequate grounding, and minimizing the proximity of noisy signals to sensitive components.

EMI Shields

569. What are EMI shields? Why/when are they needed?

EMI shields are barriers that block or attenuate electromagnetic interference. They are needed to protect sensitive components from external EMI or to prevent a device from emitting EMI that could affect other devices. Shields are typically used when there is a risk of significant interference or to meet regulatory compliance requirements.

570. How do EMI shields work?

EMI shields work by reflecting, absorbing, or redirecting electromagnetic waves. They typically consist of conductive or magnetic materials that create a barrier to prevent the passage of EMI through or around the shielded area.

571. What are EMI shields typically connected to? Why?

EMI shields are typically connected to ground. Connecting shields to ground ensures that any captured EMI is safely dissipated and does not affect the circuit. This also helps maintain the effectiveness of the shield by providing a low-impedance path to ground.

572. Where should I place EMI shields?

EMI shields should be placed around sensitive components or areas where EMI is most likely to interfere. Common locations include around high-speed digital circuits, oscillators, and power supplies. Shields should be as close as possible to the source of EMI or the sensitive circuitry.

573. What is a Faraday cage?

A Faraday cage is an enclosure made of conductive material that blocks external static electric fields and electromagnetic interference. It works by redistributing the electric charge on its surface, preventing external fields from penetrating the interior.

Vias

574. What is a via? Why are they needed?

A via is a plated hole in a PCB that allows electrical connections between different layers of the board. Vias are needed to route signals, power, or ground between different layers, enabling complex multi-layer designs.

575. If I have a signal on two separate layers, how can I connect them together?

Use a via to connect traces on separate layers. The via provides a conductive path between the layers, allowing the signal to pass through.

576. What are the different types of vias? What are their tradeoffs?

The main types of vias are through-hole vias, blind vias, and buried vias:

Through-hole vias connect all layers of the PCB and are the simplest and most common.

Blind vias connect an outer layer to one or more inner layers but do not go through the entire board.

Buried vias connect only inner layers and are not visible from the outer layers.

Tradeoffs include manufacturing complexity, cost, and potential impact on signal integrity and thermal performance.

577. If I need to via between inner layers, what type of vias would I use?

For connections between inner layers, use blind or buried vias. Blind vias connect an outer layer to one or more inner layers, while buried vias connect only inner layers and are not visible from the outside.

578. When would I want filled vias?

Filled vias are used to improve thermal and electrical conductivity, and to prevent solder wicking during assembly. They are particularly useful in high-density designs or where vias need to be filled to maintain a flat surface for subsequent processes.

579. When would I want plugged vias?

Plugged vias are used to block the via from being filled with solder or other materials. This is often done to prevent solder from wicking into the via during assembly or to create a specific electrical isolation.

580. What are some common failure modes of vias?

Common failure modes include solder wicking, thermal stress causing cracks, oxidation, and poor plating quality. These issues can lead to unreliable electrical connections or mechanical failure.

581. When do I not want to use vias?

Avoid using vias in high-speed signal paths where they can introduce inductance and signal integrity issues. Also, minimize vias in power and ground planes to reduce impedance and improve current handling.

582. Can you put a via on a surface-mount pad?

While it is technically possible to place a via on a surface-mount pad, it is generally not recommended as it can affect the reliability of solder joints and impact signal integrity. If unavoidable, ensure that the via is properly designed and does not compromise the pad’s function.

583. What happens if you have too many vias?

Too many vias can increase board cost, complexity, and signal integrity issues. Excessive vias can also cause increased inductance, reduced current-carrying capacity, and potential reliability problems.

584. What happens if you have too many vias through a power plane?

Too many vias through a power plane can reduce the plane’s effectiveness in distributing power and can create impedance issues. This can lead to increased noise, thermal hotspots, and potential power delivery problems.

585. What is "swiss cheesing" a plane?

"Swiss cheesing" refers to the practice of creating numerous holes

Testing PCBs and Design for Excellence (DFX)

Testing PCBs

586. What are some tests you can do on a PCB?

Visual Inspection: Check for obvious defects such as soldering issues, missing components, or damaged traces.

Continuity Testing: Verify that electrical connections are intact.

Insulation Resistance Testing: Ensure there is no unintended electrical path between traces.

Functional Testing: Validate that the PCB performs its intended functions under operational conditions.

In-Circuit Testing (ICT): Test the functionality of each component within the circuit.

Automated Optical Inspection (AOI): Inspect for defects using imaging systems.

587. When testing a PCB, what are you looking for?

Defects: Soldering issues, component misplacements, or manufacturing defects.

Functionality: Ensure the PCB meets design specifications and performs its intended functions.

Electrical Issues: Shorts, opens, and incorrect voltage levels.

Compliance: Adherence to design and regulatory standards.

588. What are some challenges when testing PCBs?

Complexity: Dense or multilayer designs can make access to test points difficult.

Accessibility: Some components may be difficult to test due to their location or size.

Test Coverage: Ensuring all critical parts of the circuit are tested adequately.

Automation: Integrating automated testing solutions with complex designs.

589. What are some common failure modes of a PCB?

Short Circuits: Unintended connections between conductive paths.

Open Circuits: Broken connections or disconnections.

Soldering Issues: Cold solder joints, solder bridges, or insufficient solder.

Component Failures: Damaged or defective components.

Trace Damage: Broken or damaged PCB traces.

590. How do you test ICs at the board-level?

Functional Testing: Verify the IC performs as expected in the circuit.

In-Circuit Testing (ICT): Check the IC's pins for correct voltage levels and connectivity.

Boundary Scan Testing: Use JTAG or similar methods to test IC functionality without physical probing.

591. How do you test components at the board level?

Visual Inspection: Check for placement, orientation, and soldering quality.

Electrical Testing: Measure voltage, current, or resistance to ensure proper operation.

Functional Testing: Verify the component performs its intended function within the circuit.

592. Given a PCB with x, how would you test it?

To answer this question accurately, more context about "x" is needed. Generally, you would start with a visual inspection, then proceed with continuity testing, functional testing, and if necessary, automated or specialized testing based on the PCB's complexity and function.

PCB Fabrication

593. Describe the PCB fabrication process. How are PCBs made?

Design: Create the PCB layout using CAD software.

Printing: Print the PCB design onto a laminate using photolithography.

Etching: Remove unwanted copper from the laminate to leave behind the desired circuit pattern.

Drilling: Drill holes for vias and component leads.

Plating: Plate the drilled holes with copper to make electrical connections between layers.

Solder Masking: Apply a protective layer to prevent solder from bridging traces.

Silkscreen: Print labels and markings onto the PCB.

Cutting: Cut the PCB to its final shape.

Testing: Perform electrical and functional tests to ensure quality.

594. What is panelization?

Panelization is the process of arranging multiple PCB designs on a single large panel for efficient manufacturing. It allows for the simultaneous production of multiple boards, which are later separated.

595. What are mousebites?

Mousebites are small, partial cuts or perforations in a PCB panel that help separate individual boards from the panel without fully cutting through. They allow for easy breaking apart of the boards while maintaining alignment during processing.

596. What are fiducials?

Fiducials are reference marks used on a PCB to ensure accurate alignment and placement of components during assembly and soldering. They are typically small, precise markers that help automated machinery align the PCB correctly.

Manufacturing and Assembly

597. Describe the PCB manufacturing and assembly process.

Manufacturing: Includes PCB design, fabrication (as described in 593), and testing of bare boards.

Assembly: Involves placing and soldering components onto the PCB. This includes component placement (pick-and-place), soldering (reflow or wave soldering), and final inspection.

598. What is the difference between surface-mount and through-hole components? Any tradeoffs?

Surface-Mount Components: These are placed directly onto the surface of the PCB. They allow for higher component density and are often used in high-speed or compact designs. Tradeoffs include potential difficulties with manual assembly and repair.

Through-Hole Components: These components have leads that go through holes in the PCB and are soldered on the opposite side. They are often used for components requiring mechanical strength and are easier to hand-solder. However, they take up more space and can limit the number of components on the board.

599. How are components soldered on a PCB?

Reflow Soldering: For surface-mount components, solder paste is applied, components are placed, and the PCB is heated in a reflow oven to melt the solder.

Wave Soldering: For through-hole components, the PCB is passed over a wave of molten solder that fills the holes and creates solder joints.

Manual Soldering: For small-scale or repair work, components are soldered using a soldering iron.

• How are double-sided components soldered on a PCB? What are some techniques?

Double-sided components are soldered using a combination of reflow soldering and manual soldering. The PCB is first reflow-soldered for one side, then components on the other side are soldered. Techniques include using solder paste for initial placement and reflow, followed by manual or wave soldering for additional components.

• What about mixed assembly boards?

Mixed assembly boards have both surface-mount and through-hole components. These boards typically go through a reflow soldering process for surface-mount components, followed by wave soldering for through-hole components. Careful design is required to ensure effective soldering and component placement.

600. What is pick-and-place?

Pick-and-place is an automated process used to place surface-mount components onto a PCB. Machines pick up components from feeders and place them onto the PCB according to the design. This process improves accuracy and efficiency compared to manual placement.

601. Pick-and-place machines must often traverse repetitive paths for assembly. What are some optimizations?

Optimizations include:

Efficient Path Planning: Minimize travel distance and avoid unnecessary movements.

Component Placement Efficiency: Group similar components to reduce pick-and-place cycles.

Feeder Optimization: Use automated feeders to streamline component delivery.

Machine Calibration: Regularly calibrate to maintain precision and speed.

602. What is reflow?

Reflow is a soldering process where solder paste on a PCB is melted to form solder joints. The PCB is heated in a reflow oven, causing the solder paste to flow and create connections between components and the PCB.

603. What is wave soldering?

Wave soldering is a process used for through-hole components where the PCB is passed over a wave of molten solder. The solder flows through the holes, creating solder joints on the underside of the board.

Automated Testing

604. How are PCBs tested? When does this happen?

PCBs are tested during and after manufacturing. Testing includes visual inspection, electrical tests, and functional tests. Automated testing methods are used throughout the manufacturing process to ensure quality and functionality.

605. How are PCBs tested at a large scale?

At large scale, PCBs are tested using automated methods such as In-Circuit Testing (ICT), Automated Optical Inspection (AOI), and Functional Testing (FCT). These methods are integrated into the assembly line to handle high volumes efficiently.

606. What sort of tests do PCBs undergo on the assembly line? What are the steps?

Tests include:

Visual Inspection: Check for obvious defects.

In-Circuit Testing (ICT): Test individual components and connections.

Automated Optical Inspection (AOI): Detect soldering and placement issues.

Functional Testing (FCT): Verify the PCB operates as intended.

607. What are a few test stations you would see on an electronics manufacturing/assembly line?

AOI Station: For inspecting soldering and component placement.

ICT Station: For testing electrical connections and component functionality.

FCT Station: For verifying the complete functionality of the assembled board.

Final Inspection Station: For overall quality checks and packaging.

608. What is automated optical inspection (AOI)? What can it capture?

AOI is a technique using cameras and image processing to inspect PCBs for defects such as soldering issues, missing components, or misalignments. It captures visual defects that can be difficult to detect manually.

609. What is in-circuit testing (ICT)? What can it capture?

ICT is a method of testing individual components and connections on a PCB using a bed-of-nails test fixture. It captures electrical issues such as shorts, opens, and incorrect component values.

610. What is functional testing (FCT)? What can it capture?

FCT tests the PCB under operational conditions to ensure it performs its intended functions. It captures issues related to overall circuit functionality and performance

611. What is flying probe testing? What can it capture?

Flying probe testing is a method used to test printed circuit boards (PCBs) without the need for a physical test fixture. It uses probes (needles) that move around the PCB to test connections, component values, and functionality.

What it can capture:

Continuity: Checks if all the connections between components are correctly made.

Short circuits: Detects unintended connections between traces or components.

Open circuits: Identifies missing or broken connections.

Component testing: Can measure resistances, capacitances, and diodes on the PCB.

Functional testing: Verifies if the circuit is performing its intended function.

612. What are bed-of-nails test fixtures? What can they capture?

A bed-of-nails test fixture is a type of PCB testing method that uses a fixed array of spring-loaded pins (nails) to make contact with test points on the board. The board is pressed onto the fixture, and the nails test various connections and components simultaneously.

What it can capture:

Continuity: Verifies that all connections are properly established.

Short circuits: Detects unintended connections between different circuits.

Open circuits: Identifies if any connections are missing or broken.

Functional testing: Can test the overall functionality of the circuit.

High-volume testing: It is ideal for high-speed, mass-production environments where fast, repeatable testing is required.

Design for Manufacturing (DFM)

613. How can you design a PCB for manufacturing?

Simplify Layout: Keep the design straightforward to avoid complex routing and minimize errors.

Standardize Components: Use readily available and standard parts to avoid custom or rare components.

Minimize Layers: Fewer layers can reduce manufacturing costs and complexity.

Design for Assembly: Ensure components are placed in a way that's easy for automated machines to handle and solder.

Provide Clear Documentation: Include detailed schematics and assembly instructions to guide the manufacturing process.

614. How can you ensure testability of your design at high volumes in automated testing environments?

Include Test Points: Design specific areas on the PCB where measurements and testing can be easily performed.

Design for ICT: Ensure the PCB layout allows for easy access to all necessary test points for In-Circuit Testing.

Provide Test Fixtures: Design the PCB to work with automated test fixtures and ensure it fits well with testing equipment.

615. What is component tombstoning? When can it happen? How can it be prevented?

Tombstoning occurs when one end of a component lifts off the PCB during soldering, making it stand up like a tombstone. It usually happens due to uneven solder paste application or thermal imbalance.

Prevention: Ensure even solder paste application, use proper reflow soldering profiles, and double-check component placement.

Design for Testing (DFT)

616. How can you design a PCB for testing?

Include Test Points: Add specific points on the board where measurements can be taken easily.

Design for Accessibility: Ensure components and connections are easy to access for testing.

Provide Documentation: Include detailed design and test plans to facilitate the testing process.

617. What special board design choices would you do for PCB prototyping?

Flexible Layout: Use a layout that allows for easy modifications and changes.

Add Test Points: Include extra test points for easier debugging.

Use Prototyping Boards: Consider using standard prototyping boards or modular designs to simplify testing and adjustments.

618. What are ground probes on a PCB? Why are they needed? When should they be used?

Ground Probes are specific points on a PCB used to connect ground leads during testing and debugging.

Need: They provide a stable reference point for measurements and help ensure accurate signal readings.

Use: Include ground probes in areas where precise measurements or signal integrity checks are necessary.

Test Points

619. What are test points? What are their use cases?

Test Points are specific locations on a PCB designed for connecting test equipment to measure voltages, currents, or other electrical characteristics.

Use Cases: Testing electrical signals, troubleshooting issues, and validating circuit functionality.

620. What are different types of test points?

Pad Test Points: Simple pads where a probe can make contact.

Via Test Points: Vias with exposed copper for probing.

Component Test Points: Specially designed pads or traces around components for easy measurement.

621. What are layout guidelines for test points?

Accessibility: Place test points where they are easily accessible without disrupting other components.

Spacing: Ensure sufficient space around test points for probes or testing equipment.

Visibility: Make test points clearly visible and well-documented in the design.

622. When should/shouldn't you use them?

Use Test Points: For critical signals that need to be measured or monitored, or for areas that may need frequent testing.

Avoid Test Points: On signals that are high-speed or sensitive, where probing might affect performance or introduce noise.

623. Should you use test points on high-speed or sensitive signals? Why?

Generally Avoid: Test points on high-speed or sensitive signals can introduce noise or affect signal integrity. Use special techniques if necessary, like controlled impedance probes.

624. Who/what uses test points?

Technicians: For troubleshooting and debugging.

Automated Test Equipment: For performing in-circuit tests.

Engineers: For validating and measuring circuit functionality.

625. If you're short on board space, how do you prioritize which signals should have test points?

Critical Signals: Prioritize signals that are crucial for the board's operation or that are likely to encounter issues.

Testing Needs: Consider which signals are most important for verifying the board's functionality and reliability.

Debugging and Making Measurements on PCBs

626. If there's a short on the PCB, how can it be detected, found, and root caused?

Detection: Use a continuity tester or multimeter to check for unintended connections.

Finding: Trace the circuit visually and with test equipment to locate the short.

Root Cause: Identify and fix the underlying issue, such as solder bridges or damaged traces.

627. How do you measure signals on a PCB? What are some considerations?

Use a Multimeter/Oscilloscope: For measuring voltage, current, and waveform signals.

Considerations: Ensure probes are properly connected, minimize interference, and understand the signal's characteristics.

628. How can test points be used to measure signals?

Connect Probes: Attach measurement probes to test points for accurate readings.

Isolate Signals: Use test points to isolate and measure specific signals without affecting other parts of the circuit.

629. When measuring a signal, what is the ground loop?

Ground Loop: A situation where multiple ground paths create a loop, potentially causing noise or measurement errors.

630. Why should your ground loop be as small as possible?

Minimize Noise: Smaller ground loops reduce the risk of interference and measurement errors.

631. How can you reduce your ground loop?

Shorten Connections: Keep ground paths as short and direct as possible.

Single Ground Point: Use a single ground reference point to avoid multiple ground paths.

Reworking PCBs

632. What is PCB rework?

Rework is the process of modifying or repairing a PCB after it has been assembled, usually to fix defects or make adjustments.

633. Describe any rework experience you have.

Answer depends on personal experience. Common rework tasks include replacing damaged components, correcting soldering errors, and rerouting traces.

634. What sort of rework can be done on a PCB?

Component Replacement: Removing and replacing faulty components.

Soldering Corrections: Fixing soldering issues such as bridges or cold joints.

Trace Repairs: Repairing broken or damaged PCB traces.

635. Why may rework be needed?

Defects: To correct manufacturing or assembly defects.

Design Changes: To implement design revisions or updates.

Component Failures: To replace malfunctioning components.

636. What can you do in your design to improve reworkability?

Design for Access: Ensure components and traces are accessible for easy repair.

Use Test Points: Facilitate quick troubleshooting and repairs.

Avoid Complex Layouts: Simplify the design to make rework easier.

637. Often times many power components or ground components have thermal relief pads. How does this impact reworkability?

Thermal Relief: Helps in heat dissipation during soldering but can make rework more challenging. Components with thermal relief pads may require more heat to rework, and special techniques may be needed to ensure successful soldering.

Electrical Computer-Aided Design (ECAD) Software

638. Do you have experience with Eagle/Altium/KiCad/Cadence Allegro?

Answer depends on personal experience. Mention any specific experience with these ECAD tools.

639. What experience do you have with ECAD software?

Answer depends on personal experience. Discuss your familiarity with using ECAD tools for designing PCBs, creating schematics, and performing layout work.

640. Do you have any scripting experience in ECAD?

Answer depends on personal experience. Mention any experience with scripting or automation in ECAD tools, such as writing custom scripts for design or verification tasks.

641. What are design rule checks (DRC)?

DRC are automated checks in ECAD software that ensure the PCB design meets specific rules and standards, such as spacing between traces, trace width, and pad sizes. They help catch potential issues before manufacturing.

Packaging

642. What are some common IC packages? What are their tradeoffs?

DIP (Dual In-line Package):

Pros: Easy to handle, and solder, and good for prototyping.

Cons: Takes up more space on the PCB, not suitable for high-density designs.

SMD (Surface-Mount Device):

Pros: Compact, better for high-density layouts, and automated assembly.

Cons: Harder to solder manually and can be difficult to inspect for soldering issues.

BGA (Ball Grid Array):

Pros: High-density, good thermal performance, and reliable connections.

Cons: Difficult to inspect and rework due to hidden solder balls under the package.

QFP (Quad Flat Package):

Pros: Good for medium to high-density designs and easier to handle than BGA.

Cons: Larger footprint compared to BGA and can be challenging to solder.

643. What are the tradeoffs between surface-mount and through-hole packaging?

Surface-Mount:

Pros: Smaller size, better for high-density designs, and suitable for automated assembly.

Cons: More difficult to solder manually and requires precise placement.

Through-Hole:

Pros: Strong mechanical connections, easier to solder manually, and better for components that need extra strength.

Cons: Larger footprint, less suited for high-density designs, and more costly due to additional drilling.

644. What are ball grid array (BGA) packages? What are their tradeoffs?

BGA Packages: Have a grid of solder balls on the underside of the package that connect to the PCB.

Pros: High-density, good for thermal and electrical performance, and reliable connections.

Cons: Difficult to inspect and repair due to solder balls being hidden, and requires precise soldering.

645. What is the difference between the die, wafer, and package?

Die: The small piece of silicon with the integrated circuit.

Wafer: A large, thin slice of silicon from which multiple dies are cut.

Package: The protective casing that houses the die and connects it to the PCB.

646. What are node sizes? What does "3nm process" refer to? Are transistors really that small?

Node Size: Refers to the manufacturing process technology and the smallest feature size on the chip.

"3nm Process": Indicates that the smallest features of the chip are approximately 3 nanometers. Yes, transistors are incredibly small, and advances in technology allow these tiny sizes.

647. What are package on package (PoP) chips? What are their tradeoffs?

PoP Chips: Two or more chips stacked on top of each other to save space and increase functionality.

Pros: Saves board space, can combine different types of memory and logic.

Cons: Can have thermal management issues and higher complexity in assembly.

648. How do PoP chips impact thermals?

Impact: Stacking chips can create heat management challenges, as heat generated by one chip can affect the other. Good thermal design and management are required to avoid overheating.

649. Which chips are often PoP? Why?

Often PoP: Memory chips (e.g., DRAM) and processors, often used in mobile devices and compact electronics to save space and integrate components.

650. Describe any yield considerations when fabricating chips.

Yield Considerations: Factors like defects in the wafer, processing variations, and imperfections can affect the number of functional chips produced. Higher yields mean more usable chips per wafer, which is crucial for cost-effectiveness.

651. Why can't ginormous chips, single die be made? What are some challenges and workarounds?

Challenges: Large chips can have issues with heat dissipation, higher chances of defects, and difficulty in manufacturing.

Workarounds: Use multiple smaller chips or chiplets combined to achieve desired functionality while managing heat and manufacturing complexities.

652. What are chiplets?

Chiplets: Small, modular chips that can be combined to form a larger, more complex system. They help manage manufacturing challenges and allow for more flexibility in design.

653. What are 3D chips?

3D Chips: Chips where multiple layers of integrated circuits are stacked vertically. This design can improve performance and reduce space but requires advanced manufacturing techniques.

654. What are bond wires?

Bond Wires: Thin wires used to connect the die inside a chip to the external leads or pads on the chip package.

655. Bond wires are often very thin but sometimes may need to carry lots of current. How do you reconcile this?

Reconciliation: Thin wires are typically used for signals or low-current connections. For higher currents, designs might use multiple wires or alternative techniques like using larger bond pads to handle the power more effectively.

656. What are flip chips?

Flip Chips: Chips where the die is flipped upside down and mounted directly on the PCB with solder bumps. This approach allows for higher density and better performance but is more complex to manufacture.

657. Why would the ground potential of a chip vary?

Variation Reasons: Differences in power supply quality, variations in current flow, and layout issues can cause variations in the ground potential, leading to noise and performance issues.

IC Fabrication

658. How are ICs made?

Process: ICs are made by growing and processing a silicon wafer, applying layers of materials, etching patterns, and packaging the processed dies into finished chips.

659. What is a wafer?

Wafer: A thin, round slice of silicon used as the base for fabricating integrated circuits. Multiple chips are manufactured on a single wafer.

660. What is the process from wafer to chip?

Process: Start with a silicon wafer, apply layers of material, pattern the circuit designs using photolithography, etch away excess material, and finally, cut the wafer into individual chips which are then packaged.

661. What is photolithography?

Photolithography: A process used to transfer a circuit pattern onto the wafer using light-sensitive chemicals and masks.

662. What is etching?

Etching: The process of removing material from the wafer to create the desired circuit patterns after photolithography.

663. What is atomic layer deposition?

Atomic Layer Deposition (ALD): A technique to deposit very thin, uniform layers of material one atom at a time, used to build precise structures on the wafer.

664. Some feature sizes may be in the nm range, but the wavelengths of light used may not be in that same range. How are these small feature sizes created?

Creating Small Features: Advanced techniques like extreme ultraviolet (EUV) lithography are used, which can create very small features even when the light wavelengths are larger by using special methods to project and focus light.

Signals

665. What is the difference between an analog signal and a digital signal?

Analog Signal: Varies continuously and can represent a range of values (e.g., sound waves).

Digital Signal: Represents information as discrete values or binary code (0s and 1s).

666. What is the Fourier Transform?

Fourier Transform: A mathematical tool that converts a signal from its time domain into its frequency domain, showing how much of each frequency is present in the signal.

667. What is the difference between the Fourier Transform and Fourier Series?

Fourier Transform: Analyzes signals of any duration and can handle both continuous and discrete signals.

Fourier Series: Breaks down periodic signals into a sum of sine and cosine functions, used for signals that repeat over time.

668. If I want to measure a sine wave, what should be the sample rate?

Sample Rate: Should be at least twice the highest frequency of the sine wave to accurately capture the signal (according to the Nyquist rate).

669. What is the Nyquist rate of a signal? Why is it important?

Nyquist Rate: Twice the highest frequency of the signal. It's important to avoid aliasing and ensure that the signal is sampled accurately.

670. What is aliasing?

Aliasing: Occurs when a signal is sampled at a rate lower than its Nyquist rate, causing different signals to become indistinguishable and distort the representation.

Square Waves

671. What is a square wave? What are its properties?

Square Wave: A type of waveform that switches between a high and a low state with abrupt transitions, forming a square-like shape.

Properties:

- Frequency: Determines how often the waveform repeats per second.
- Duty Cycle: The ratio of the time the waveform is in the high state to the total period of the waveform.
- Symmetry: Ideal square waves are symmetric, with equal time spent in high and low states.

672. What is PWM? What are its characteristics?

PWM (Pulse Width Modulation): A technique for controlling power delivery by varying the width of the pulses in a signal while keeping the frequency constant.

Characteristics:

Duty Cycle: The proportion of the period where the signal is high.

Frequency: The rate at which the pulses repeat.

Control: Adjusts the average power delivered to a load.

673. Draw spectral content of square wave/pulse train.

Spectral Content: A square wave contains a fundamental frequency and odd harmonics of that frequency (e.g., 3rd, 5th, 7th harmonics). In a frequency domain graph, it would show a peak at the fundamental frequency and then spikes at odd multiples of this frequency, with decreasing amplitude.

674. Why are square waves so noisy? What can you do to decrease noise? What about decreasing noise at a particular frequency content?

Why Noisy: Square waves have sharp transitions that create high-frequency components (harmonics), leading to noise.

To Decrease Noise:

Filtering: Use low-pass filters to smooth out high-frequency components.

Softening Edges: Use slower rise and fall times to reduce the generation of high-frequency harmonics.

At Specific Frequencies: Implement targeted filters or shielding to attenuate noise at those frequencies.

675. What is the highest frequency component of a real vs ideal square wave?

Ideal Square Wave: Theoretically contains an infinite number of harmonics.

Real Square Wave: Limited by practical factors like rise and fall times, which cut off higher frequency components.

676. What is the rule of thumb for the highest frequency content of a real square wave?

Rule of Thumb: The highest significant frequency component is typically around 5 to 10 times the fundamental frequency, depending on the rise and fall times.

677. What are the tradeoffs of sharper edges on a square wave?

Tradeoffs:

Pros: Can be useful for high-speed digital signals where precise timing is critical.

Cons: Generates more high-frequency noise and can lead to signal integrity issues due to increased electromagnetic interference (EMI).

678. How does the frequency content of sharp vs not sharp square waves?

Sharp Square Waves: Have a broad frequency spectrum with significant higher-order harmonics due to their rapid transitions.

Not Sharp Square Waves: Have a narrower frequency spectrum with fewer high-frequency components because the edges are more gradual.

Differential Signaling

679. What is a differential pair? How does it work?

Differential Pair: Consists of two traces carrying the same signal but with opposite polarities. One trace carries the positive signal, and the other carries the negative signal.

How It Works: The receiver measures the voltage difference between the two traces. This helps cancel out noise that affects both traces equally, improving signal integrity.

680. What is the difference between a single-ended and differential signal?

Single-Ended Signal: Uses one trace with reference to ground. It's more susceptible to noise since the signal is referenced to a single point.

Differential Signal: Uses two traces with opposite polarities. The signal is defined by the difference between the traces, which helps reject common-mode noise.

681. When are they often used?

Differential Signals: Often used in high-speed data communication (e.g., USB, Ethernet) and environments with significant electromagnetic interference (EMI) to improve noise immunity and signal integrity.

Single-Ended Signals: Common in lower-speed, less noise-sensitive applications.

682. How do they improve noise immunity?

Improvement: Differential signaling rejects common-mode noise by comparing the difference between two signals. Noise that affects both lines equally gets canceled out in the differential measurement.

683. What does the common-mode voltage refer to?

Common-Mode Voltage: The average voltage level shared by both traces of a differential pair, referenced to ground. It is the same on both traces and does not affect the differential signal.

684. What does the differential voltage refer to?

Differential Voltage: The voltage difference between the two traces of a differential pair. It represents the actual signal being transmitted and is used to detect the intended information.

Signal Integrity/Power Integrity (SI/PI)

685. What is SI/PI? Why is it important?

SI (Signal Integrity): Refers to maintaining the quality of electrical signals as they travel through a circuit. Good SI ensures that signals arrive at their destination without errors, distortion, or loss.

PI (Power Integrity): Involves ensuring that the power supplied to the components is stable and clean. It deals with managing noise, voltage drops, and ensuring that power delivery is reliable.

Importance: Both SI and PI are crucial because they affect the performance and reliability of electronic circuits. Poor signal integrity can cause data errors, while poor power integrity can lead to unstable operation or even damage components.

686. Do you have any SI/PI simulation experience?

Explanation: Yes, I have experience with SI/PI simulations, which involves using specialized software to predict and analyze how signals and power behave in a circuit design. This helps identify potential issues and optimize the design for better performance before building the physical hardware.

Signal Integrity (SI)

687. How to characterize the quality of a signal?

Signal quality is characterized by rise/fall times, overshoot, undershoot, jitter, ringing, and signal integrity metrics like noise margin and eye diagrams.

688. What is ground bounce?

Ground bounce occurs when rapid switching causes fluctuations in the ground voltage level, resulting in noise in nearby circuits.

689. What happens if the signal rising edge is too slow? Too fast?

Too slow: Increased susceptibility to noise and timing issues. Too fast: Can cause ringing, overshoot, and signal reflection.

690. What is signal monotonicity?

Monotonicity means a signal increases or decreases without reversing direction during a transition (no oscillations or ripples).

691. Why is monotonicity during a signal transition important?

Monotonicity ensures correct timing and prevents misinterpretation of signal edges by logic circuits.

692. What is hysteresis and why is it used?

Hysteresis adds a threshold difference between the rising and falling edges of a signal to prevent noise from causing false triggering, improving signal stability.

693. What are setup and hold times? What happens if they're violated?

Setup time: Time before the clock edge when data must be stable. Hold time: Time after the clock edge when data must remain stable. Violation: Leads to metastability and unreliable data.

694. What is crosstalk?

Crosstalk is unwanted interference caused by the coupling of signals between adjacent traces or wires, degrading signal quality.

695. What is jitter?

Jitter is the deviation in signal timing from its ideal position, causing timing errors and performance degradation in high-speed systems.

696. What is ringing?

Ringing is the oscillation of a signal after a transition, caused by impedance mismatches or inductive effects, leading to signal integrity issues.

Differential Pairs (SI Perspective)

Diff pairs from SI perspective, as opposed to board layout/design or signals perspective.

Signal Integrity (SI)

697. How do you characterize SI of a differential pair?

Characterization: You measure how well a differential pair transmits signals without degradation. This includes checking:

Signal Quality: Look at waveform shapes and timings.

Crosstalk: Ensure the signals don't interfere with each other.

Differential Impedance: Verify that the impedance of both traces is matched.

698. What is an eye diagram? What does it represent?

Eye Diagram: A graphical representation of a digital signal's quality, showing multiple bits over time. It's like an "eye" shape when viewed on an oscilloscope.

Represents: The clarity of the signal, showing timing margins, noise, and signal distortions.

699. What happens if the eye diagram is too small? What is often used to represent this?

Too Small Eye: Indicates poor signal quality with significant timing errors or noise. It suggests data corruption or reliability issues.

Representation: Often represented by a "mask" or "masking" where the eye's shape overlaps the defined safe areas.

700. What is the mask of an eye diagram? What happens if some of the signal enters the mask?

Mask: A predefined boundary that the eye diagram should stay within.

Entering Mask: If the signal crosses into the mask area, it means there is too much noise or distortion, and the signal quality is inadequate.

701. What can you do to expand the eye?

Expand Eye: Improve signal quality by:

Reducing Noise: Minimize external interference.

Improving Timing: Optimize clock and data timing.

Signal Conditioning: Use equalization or preemphasis to clean the signal.

702. What can you do to ensure the eye stays out of the mask?

Stay Out of Mask: Improve signal integrity by:

Proper Design: Ensure good PCB layout and impedance matching.

Signal Conditioning: Apply techniques like equalization to reduce distortions.

Minimizing Interference: Use shielding and proper grounding.

703. How is timing jitter and amplitude noise illustrated on an eye diagram?

Timing Jitter: Causes the eye to become narrower or show irregularities in the horizontal axis.

Amplitude Noise: Results in a more closed or irregular eye shape, affecting vertical consistency.

Power Integrity (PI)

Lots of crossover with Power Electronics section.

704. How do you characterize quality of power supply?

Characterize Quality: Measure the stability and cleanliness of the power supply by checking:

Ripple: Fluctuations in the voltage.

Noise: High-frequency interference.

Voltage Regulation: How well the supply maintains its voltage.

705. What is power supply ripple? Where does it come from?

Power Supply Ripple: Small, periodic fluctuations in the voltage output of a power supply.

Source: Comes from the conversion process (AC to DC) and is influenced by load changes and filtering.

706. What is power sequencing? Why is it important?

Power Sequencing: The order and timing in which power rails are turned on or off.

Importance: Ensures that components power up in the correct sequence to avoid damage and ensure proper startup.

707. What are power supply spikes? What could be their cause and how can they be captured?

Power Supply Spikes: Sudden, brief increases in voltage.

Causes: Sudden changes in load, switching components, or transient responses.

Capture: Use oscilloscopes or specialized measurement tools to detect and analyze them.

Transmission Lines

More on generic transmission line (TL) and high-speed design theory, as opposed to its implementation/analysis in the context of PCB design which is in its own section: Printed Circuit Boards (PCBs) > Design/Layout/Routing > High-Speed Signals

TL Theory

708. What is TL theory? Why is it needed?

TL Theory: Study of how electrical signals travel through transmission lines (like traces on a PCB).

Need: To design circuits that transmit signals accurately without reflections or loss.

709. When is TL analysis valid?

Valid: When dealing with high-frequency signals or long traces where the signal's wavelength is comparable to the trace length.

710. What is characteristic impedance?

Characteristic Impedance: The resistance of a transmission line to the flow of current. It depends on the line's physical dimensions and materials.

711. What are reflections? Why do I care about them?

Reflections: Occur when signals bounce back from impedance mismatches.

Importance: Reflections can cause signal distortion and loss, affecting the performance of the circuit.

712. Draw some signals with/without reflections.

Without Reflections: A clean signal with no distortion.

With Reflections: A signal with additional noise or multiple overlapping waves.

713. What causes reflections?

Causes: Impedance mismatches along the transmission line, such as abrupt changes in width or material.

714. What is impedance matching? Why do I care about it and when should it be used?

Impedance Matching: Ensuring that the impedance of the transmission line matches the source and load impedance.

Importance: To minimize reflections and signal loss, it's crucial for high-speed or high-frequency signals.

715. How does R, L, and C impact characteristic impedance? How does that differ on a microstrip vs coax TL?

Impact:

R (Resistance): Can cause signal loss.

L (Inductance) and C (Capacitance): Determine the impedance value and how it varies with frequency.

Microstrip vs Coax: Microstrips have different impedance characteristics due to their geometry and dielectric material compared to coaxial cables.

716. What is the difference and tradeoffs of series vs parallel termination?

Series Termination: Places a resistor in series with the signal line.

Pros: Reduces reflections; simple to implement.

Cons: Can affect signal amplitude.

Parallel Termination: Places a resistor to ground at the end of the line.

Pros: Provides better impedance matching; reduces reflections more effectively.

Cons: May increase power consumption.

717. What is the difference and tradeoffs of source vs sink termination/matching?

Source Termination: Matching impedance at the source end.

Pros: Helps with signal integrity at the point of signal generation.

Cons: May not fully eliminate reflections if the line is long.

Sink Termination: Matching impedance at the load end.

Pros: Effective for minimizing reflections at the receiving end.

Cons: Requires careful placement and can be more complex.

Ideal vs Real TLs

718. Draw real TL distributed model. What are its components on a real TL? How do they differ between ideal vs real?

Real TL Model:

Components: Includes resistive (R), inductive (L), and capacitive (C) elements distributed along the line.

Ideal TL: Assumes no losses or distortions.

Real TL: Includes losses due to resistance, parasitic inductance, and capacitance.

719. How can I control the characteristic impedance of a TL?

Control Impedance:

Design: Adjust the width and spacing of traces.

Material: Use different dielectric materials or thicknesses.

Layer Stackup: Change the arrangement of layers and grounding.

720. What are some standard termination/characteristic impedances?

Standard Impedances:

50 Ohms: Common for RF and high-speed signals.

75 Ohms: Used for video and some data applications.

721. Describe the parasitic R, L, and C components that impact characteristic impedance.

Parasitic R (Resistance): Causes signal attenuation and power loss.

Parasitic L (Inductance): Adds impedance and can affect signal timing.

Parasitic C (Capacitance): Affects signal speed and can cause coupling between traces.

722. What are sources of loss and distortion in a TL? How can they be minimized or actively compensated for?

Sources of Loss:

Resistance: Causes signal attenuation.

Dielectric Losses: Energy loss in the insulating material.

Reflections: Due to impedance mismatches.

Minimization: Use proper impedance matching, high-quality materials, and good design practices.

723. What are dielectric losses? Where do they come from and how does it vary with frequency?

Dielectric Losses: Energy lost in the dielectric material between conductors.

Source: Due to the material's inherent properties.

Frequency Dependence: Losses increase with higher frequencies.

724. How does a TL's insertion impedance vary with respect to frequency?

Insertion Impedance: Changes with frequency due to the combined effects of resistance, inductance, and capacitance.

Higher Frequencies: Impedance may increase due to inductive effects.

725. What is the skin effect? Where does it come up in TLs?

Skin Effect: The tendency for AC current to flow near the surface of a conductor, reducing effective cross-sectional area.

In TLs: Leads to higher resistance at high frequencies, affecting signal integrity.

726. What is preemphasis? Why is it used? How can it be implemented?

Preemphasis: Technique to boost high-frequency components of a signal before transmission.

Purpose: Compensates for signal loss in the transmission line.

Implementation: Use electronic circuits to apply a frequency-dependent gain

727. Draw the time-domain and frequency-domain spectrum of a signal with/without preemphasis.

Time-Domain (Signal Waveform):

Without Preemphasis: The signal has uniform amplitude across different frequencies. You may see a flat waveform where high-frequency components are not distinctly different from low-frequency components.

With Preemphasis: The waveform may show that high-frequency components are more pronounced before transmission, as they have been boosted.

Frequency-Domain (Spectrum):

Without Preemphasis: The spectrum has a relatively flat response across frequencies, indicating equal amplitude for all frequency components.

With Preemphasis: The spectrum shows an increase in amplitude at higher frequencies. This boost helps counteract the loss of high frequencies during transmission, making the signal more robust.

Embedded Systems

728. What are some components/subsystems you'd find in an embedded system?

Microcontroller or Microprocessor: The brain of the system.

Memory: For storing data and instructions (RAM, ROM, Flash).

I/O Interfaces: For communication with external devices (e.g., GPIO pins, UART, SPI, I2C).

Power Supply: Provides the necessary power to the system.

Clock: Provides timing signals to synchronize operations.

Sensors and Actuators: For interacting with the physical world (e.g., temperature sensors, motors).

729. What are some examples of embedded systems?

Home Appliances: Smart refrigerators, washing machines.

Automotive Systems: Engine control units, infotainment systems.

Consumer Electronics: Smartwatches, fitness trackers.

Industrial Equipment: Robotics controllers, manufacturing automation systems.

Medical Devices: Heart rate monitors, infusion pumps.

Microcontrollers

730. What are some common microcontrollers/microprocessors?

Microcontrollers: Arduino (ATmega series), ESP32, PIC microcontrollers, STM32.

Microprocessors: Intel Core processors, ARM Cortex-A series, AMD Ryzen.

731. What is the difference between a microcontroller and a microprocessor?

Microcontroller: Integrates CPU, memory, and I/O peripherals into a single chip. Used in embedded systems where cost and size are critical.

Microprocessor: Focuses on high processing power and typically requires external components for memory and I/O. Used in computers and complex applications.

732. What are the blocks of a microcontroller and microprocessor?

Microcontroller: CPU, RAM, ROM/Flash, I/O ports, timers, ADC/DAC.

Microprocessor: CPU, cache memory, RAM, external memory interfaces, I/O interfaces.

733. How does a microcontroller interact with external peripherals?

Through I/O Pins: They can read from or write to external sensors and devices.

Communication Protocols: Such as UART, SPI, I2C to exchange data with other components.

734. When do you use MCU vs ASIC vs FPGA?

MCU (Microcontroller): For general-purpose tasks with moderate complexity and low cost.

ASIC (Application-Specific Integrated Circuit): For high-performance and specialized tasks, with high volume production.

FPGA (Field-Programmable Gate Array): For customizable hardware designs that need to be modified or updated after manufacturing.

735. How do you spec an MCU? What specs do you look for?

Clock Speed: Determines how fast the MCU operates.

Memory: Amount of RAM and Flash storage.

I/O Ports: Number and type of input/output pins.

Communication Interfaces: Available UART, SPI, I2C.

Power Consumption: Important for battery-powered devices.

736. What is a reset signal? What does it do? Why is it useful?

Reset Signal: A signal that initializes or restarts the microcontroller.

Function: It clears the system's memory and brings the microcontroller to a known starting state.

Usefulness: Ensures the system starts correctly and consistently after power-up or if an error occurs.

Communication Buses

737. What are the three common communication protocols? How do they (I2C, SPI, UART) work?

I2C (Inter-Integrated Circuit): Uses a two-wire system (SDA for data, SCL for clock) to connect multiple devices with a master-slave configuration.

SPI (Serial Peripheral Interface): Uses four wires (MISO, MOSI, SCK, and SS) for high-speed communication between a master and multiple slaves.

UART (Universal Asynchronous Receiver-Transmitter): Uses two wires (TX and RX) for asynchronous communication between two devices.

738. What is the difference between I2C and SPI?

I2C: Uses two wires (SDA and SCL), slower speeds, supports multiple devices with unique addresses.

SPI: Uses four wires (MISO, MOSI, SCK, SS), faster speeds, more straightforward but requires more pins.

739. Which is faster? Why?

SPI: Generally faster than I2C because it has higher clock speeds and doesn't require address recognition.

740. How do different chips talk to each other?

Using Communication Protocols: Chips communicate via protocols like I2C, SPI, or UART, which define how data is transmitted and received.

741. What is the difference between asynchronous vs synchronous communication? What are the tradeoffs?

Asynchronous Communication: Doesn't use a clock signal; relies on start and stop bits. Simpler but less efficient for high-speed data.

Synchronous Communication: Uses a clock signal to synchronize data transmission. More efficient but requires clock lines and synchronization.

742. What are logic levels?

Logic Levels: Represent the different states in digital circuits, usually denoted as HIGH (1) and LOW (0).

743. What are some common logic levels?

TTL (Transistor-Transistor Logic): Typically 0V for LOW and 5V for HIGH.

CMOS (Complementary Metal-Oxide-Semiconductor): Can be 0V for LOW and 3.3V or 5V for HIGH, depending on the system.

I2C

744. What is I2C? What does it stand for?

I2C: Stands for Inter-Integrated Circuit. It's a protocol for connecting multiple devices using two wires.

745. How does it work? Explain its operation.

Operation:

Two Wires: SDA (Serial Data) for data and SCL (Serial Clock) for clock.

Master-Slave: One master controls the communication, and multiple slaves can be connected.

Addressing: Each slave device has a unique address.

Data Transfer: Data is sent in packets, with the master initiating and controlling the communication.

746. How interface with slaves?

Using Addresses: The master sends the address of the slave it wants to communicate with before data transfer.

747. How many slaves can there be?

Up to 127: Address space allows for 127 unique slave addresses (7-bit address format).

748. When would you need more masters?

Multiple Control Points: When you need more than one device to control or manage communication with slaves.

749. Explain address/data frames.

Address Frame: Contains the address of the slave device.

Data Frame: Contains the actual data being transmitted.

750. How can you change a slave's address?

By Setting Address Pins: Some devices have pins to set the address manually.

Through Configuration Registers: Some devices allow changing the address via software.

751. Explain NACK/ACK.

ACK (Acknowledgment): Indicates successful reception of data.

NACK (Not Acknowledged): Indicates that data was not successfully received or processed.

752. Explain clock stretching.

Clock Stretching: Allows a slave device to slow down the clock signal to give it more time to process data.

Physical Layer

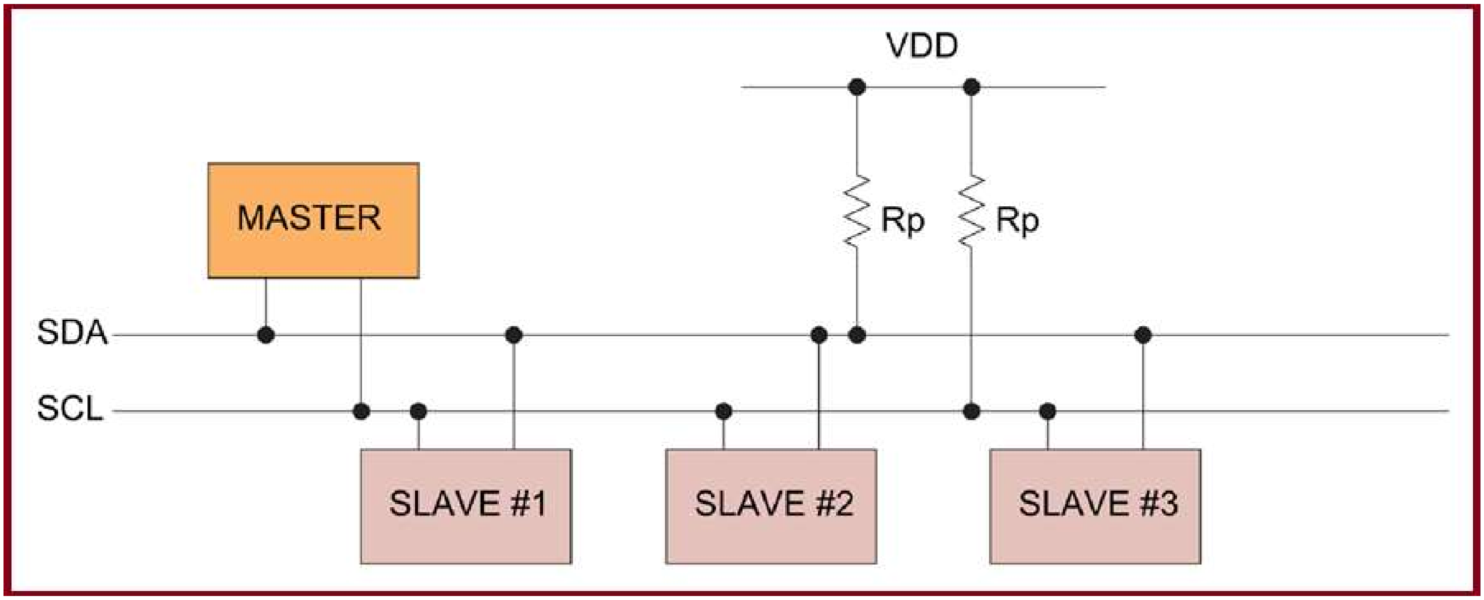
753. How many pins/wires are needed? What if I have more slaves?

I2C: Needs 2 pins/wires: SDA (data line) and SCL (clock line). You can connect multiple slaves using the same two wires, but you need to address each slave uniquely.

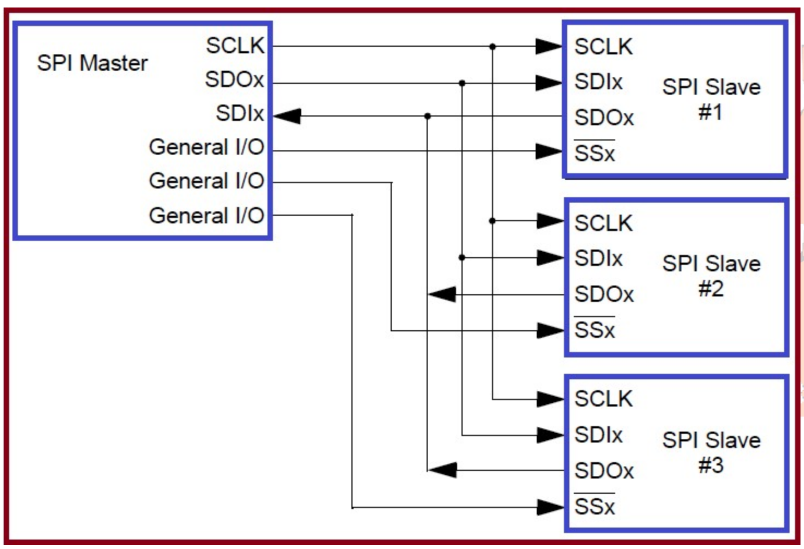
SPI: Needs 4 pins/wires: MISO (Master In Slave Out), MOSI (Master Out Slave In), SCK (Clock), and SS (Slave Select). For multiple slaves, you need additional SS lines or a daisy-chain setup.

754. How do I make the connections? Draw it out.

I2C: Connect SDA and SCL lines from the master to each slave device. All devices share these two lines.



SPI: Connect MISO, MOSI, and SCK lines from the master to each slave device. Each slave device needs its own SS line.



755. Why are pullups needed? How do you spec them?

Pullups: Needed on the I2C lines (SDA and SCL) to ensure they return to a high state when not driven low by any device.

Spec: Typically 4.7kΩ or 10kΩ resistors, depending on bus speed and capacitance. Lower values for faster speeds and higher capacitance.

756. What is bus capacitance?

Bus Capacitance: The total capacitance of the lines and connected devices on the I2C bus. It affects the speed and reliability of the communication.

757. How would bus capacitance change? What impacts it?

Change: Increases with more connected devices or longer wires. Higher capacitance slows down signal transitions and can cause communication errors.

758. How do you increase rise time on I2C?

Increase Rise Time: Use larger pullup resistors to slow the rate at which the signal rises. This is often done to match the bus capacitance and achieve reliable communication.

759. Why would rise time be too slow on I2C?

Slow Rise Time: May be due to overly large pullup resistors, high bus capacitance, or excessive length of the I2C bus lines.

760. If the logic low state of the signal doesn't reach the logic-low threshold of the device, what could be happening?

Possible Issues: There could be a problem with the pull-up resistors, excessive resistance in the circuit, or a weak driver. It may cause unreliable communication.

761. What are advantages of open-drain output drivers?

Advantages: Allows multiple devices to share a single line, as devices only pull the line low and let pull-up resistors bring it high. This is useful for I2C bus communications.

762. What are typical logic levels?

TTL Logic Levels: 0V for LOW and 5V for HIGH.

CMOS Logic Levels: Typically 0V for LOW and 3.3V or 5V for HIGH, depending on the system.

763. What are typical clock speeds?

I2C: Typically 100kHz (standard mode) or 400kHz (fast mode), with some supporting up to 1MHz (fast-mode plus).

SPI: Can vary widely; typical speeds are 1MHz to 10MHz, but some systems can go higher.

SPI

764. What is SPI? What does it stand for?

SPI: Stands for Serial Peripheral Interface. It’s a protocol for high-speed data transfer between a master and one or more slave devices.

765. How many slaves can there be?

Slaves: You can connect multiple slaves, but each requires its own SS line unless using daisy-chaining.

766. How to interface with slaves?

Through SS Lines: Use individual Slave Select (SS) lines to select each slave device for communication.

767. How do I make the connections?

Connections:

MOSI: Connects from master to all slaves for data out.

MISO: Connects from all slaves to master for data in.

SCK: Connects from master to all slaves for clock.

SS: Connects from master to each slave’s SS pin.

768. How does daisy-chaining work?

Daisy-Chaining: Connects the MISO of one slave to the MOSI of the next slave. Only one SS line is needed for the first slave; each subsequent slave is selected in sequence.

769. What are typical logic levels?

Logic Levels: Typically 0V for LOW and 3.3V or 5V for HIGH, depending on the system.

770. What are typical clock speeds?

SPI: Can range from 1MHz to 10MHz, with some systems going higher depending on the requirements.

UART

771. What is UART? What does it stand for?

UART: Stands for Universal Asynchronous Receiver/Transmitter. It’s a protocol for asynchronous serial communication.

772. How does it work?

Operation: Transmits data in a series of bits, with start and stop bits to frame each data byte. It doesn't use a clock signal; timing is handled by the baud rate.

773. How many wires does it have?

Wires: Typically 2 wires: TX (transmit) and RX (receive).

774. How do I make the connections?

Connections:

TX: Connects from the transmitting device to the RX of the receiving device.

RX: Connects from the receiving device to the TX of the transmitting device.

775. What are some standard communication rates?

Rates: Common rates include 9600, 19200, 38400, 57600, and 115200 baud.

776. How many devices can talk to each other? What is the direction of communication?

Devices: Typically, only two devices can communicate directly via UART. The communication is point-to-point (one transmitter and one receiver).

777. Is it synchronous or asynchronous?

Asynchronous: UART is asynchronous because it does not use a clock signal for data transmission.

778. Since there's no clock, how is data read?

Data Reading: The receiver uses the baud rate to determine when to sample the incoming data bits.

779. What is oversampling? How is it implemented? Why is it needed? What are some common oversample rates for UART?

Oversampling: Sampling the incoming signal more frequently than the baud rate to ensure accurate data capture.

Implementation: Typically involves sampling at rates 16x or 32x the baud rate to improve timing accuracy.

GPIOs

780. What is a GPIO? What does it stand for and what are its use cases?

GPIO: Stands for General Purpose Input/Output. It's a pin on a microcontroller that can be configured as either an input or an output for interfacing with other hardware.

781. What does "GP" refer to?

GP: Refers to "General Purpose," indicating that these pins can be used for a variety of tasks, not just for specific functions.

782. What can I do if I don't have enough GPIOs?

Options:

Use GPIO Expanders: Chips that increase the number of GPIOs.

Multiplexing: Share GPIO pins among multiple functions.
Shift Registers: Add more outputs using serial-to-parallel conversion.

Design/Implementation

A lot on GPIO design is very applicable, more in-depth questions are covered in the CMOS and Amplifiers subsections of Analog Electronics. These questions will just be surface-level that are more applicable in embedded applications.

783. Draw a GPIO circuit. How does it work?

A typical GPIO circuit consists of a pin connected to a switch or sensor and controlled by the microcontroller (MCU). You can configure the pin as either input (to read signals) or output (to send signals).

How it works: When set as input, the GPIO reads the voltage on the pin to determine a HIGH (1) or LOW (0) state. When set as output, the MCU sets the pin to HIGH (1) or LOW (0) to control an external device (like an LED).

784. What are the two most common types of GPIO implementations? What are their tradeoffs?

Push-pull: Directly drives the pin HIGH or LOW. It's fast and provides better power control but needs more current.

Open-drain: Can only pull the pin LOW; an external resistor pulls it HIGH. It's useful for shared lines but slower and requires external components.

785. What does open-drain/push-pull mean?

Open-drain: The GPIO can only pull the pin to ground (LOW) and requires an external pull-up resistor to get HIGH.

Push-pull: The GPIO can drive the pin both HIGH and LOW, controlling both states directly.

786. When are pull-up/pull-down resistors necessary? How do you size them?

They're needed when you want to ensure a default HIGH or LOW state when the GPIO is not actively driven.

Size them by balancing power consumption and signal speed. Typically, values range from 1kΩ to 100kΩ depending on the circuit.

787. What are some typical configurable GPIO settings?

Input/Output mode: Whether the pin reads data or sends signals.

Pull-up/Pull-down resistors: Whether to enable internal resistors to avoid floating states.

Drive strength: The amount of current the pin can source or sink.

Interrupt trigger: Configuring the pin to trigger on rising/falling edges or both.

788. What is bus contention?

Bus contention happens when two or more devices try to control a shared line (like GPIO) at the same time, leading to conflicts. This can cause damage or malfunction.

Timers in Microcontrollers

789. What are timers in a microcontroller?

Timers are hardware peripherals in microcontrollers that keep track of time and can generate periodic events or delays.

790. What can you do with timers?

Generate precise delays

Measure time intervals

Generate PWM signals

Control time-sensitive tasks like motor control

791. How do you keep track of time accurately in a microcontroller?

Use timers with a stable clock source (like a crystal oscillator) and configure them properly to generate accurate intervals.

792. How do timers work? How are they implemented?

Timers count clock cycles and trigger events when they reach a certain value. You can configure them to operate in different modes (like counting up or down).

793. What are some applications?

Time delays

Pulse Width Modulation (PWM) for motor control

Real-time clocks

Event counters

Interrupts in Microcontrollers

794. If you want to do something right when an external event occurs, what can you do?

Use an interrupt, which immediately stops the normal code execution to handle the event.

795. What are interrupts in a microcontroller?

Interrupts are signals that tell the MCU to stop its current task and handle an urgent event, like a button press or sensor trigger.

796. How do they work? How are they implemented?

When an interrupt occurs, the MCU pauses its main program, saves its current state, and runs a specific function (ISR). Once the ISR is done, it resumes normal operation.

797. Describe how ISRs work at the hardware and software levels.

Hardware level: The interrupt signal tells the MCU to stop the current operation.

Software level: The MCU runs the Interrupt Service Routine (ISR), a small function designed to quickly handle the event.

798. What are some applications?

Handling sensor data

Responding to user inputs (like button presses)

Managing time-critical tasks

799. What can trigger ISRs?

External events like button presses, timers, or communication peripherals like UART, SPI, or I2C.

800. What are some limitations of interrupts?

If the ISR is too long, it can delay other tasks.

Multiple interrupts at the same time can cause conflicts or missed events.

801. What is an Interrupt Service Routine (ISR)?

The ISR is a special function that runs when an interrupt occurs. It handles the event quickly and returns control back to the main program.

802. What are some rules of thumb when implementing ISRs?

Keep them short and efficient.

Avoid using complex logic or long delays.

Be careful when accessing shared resources.

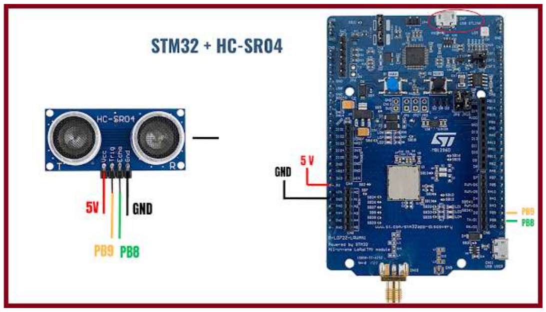
803. What happens if an ISR gets called while in an ISR? How can this be avoided?

This is called nested interrupts. It can be avoided by disabling interrupts temporarily while in the ISR or using priority levels for interrupts.

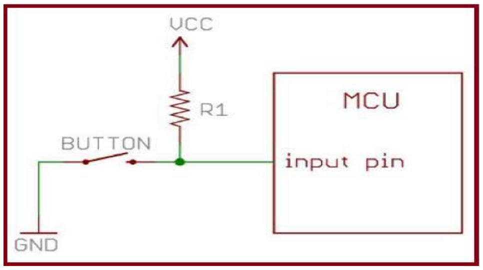
Peripherals

A lot of the inductive drive questions have good overlap with the Single-Phase Inductive Drives section (Power Electronics > Inductive Loads > Single-Phase Inductive Drives).

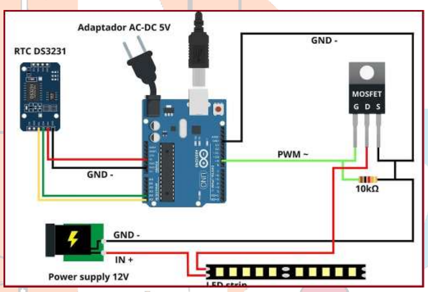
804. Draw a circuit to connect a sensor to a microcontroller.



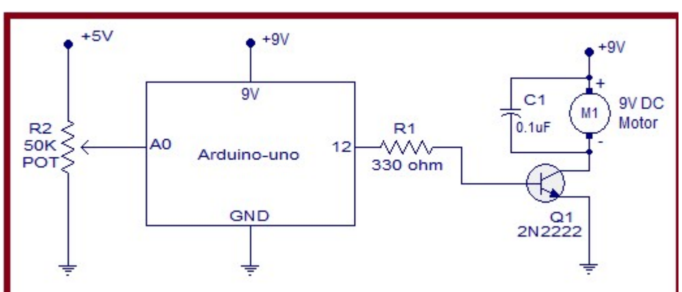
805. Draw a circuit to connect a button to a microcontroller.



806. Draw a circuit to drive a LED from a microcontroller. What about a high-power LED? What if I want to change the brightness?



807. Draw a circuit to drive a motor from a microcontroller. What if isolation is required? What if I want to change the speed?



PWM motor speed control using Arduino

ADC

Most of my ADC/DAC questions have been in the context of embedded systems, not too much on analog design or anything.

ADC (Analog-to-Digital Converter)

808. How does an ADC work?

ADC: Converts an analog signal (like a voltage) into a digital value that a microcontroller can understand. It samples the analog signal at regular intervals and translates each sample into a digital number.

809. How does a flash ADC work?

Flash ADC: Uses a parallel array of comparators to quickly convert an analog input into a digital output. It's very fast because it compares the input to many reference voltages simultaneously.

810. How does a SAR ADC work?

SAR ADC: Uses a Successive Approximation Register to convert an analog signal. It guesses the output, compares it to the input, and adjusts the guess until it matches the input. This process is slower than flash ADC but often more precise.

811. Design an ADC.

Design: Choose between flash or SAR depending on speed and accuracy needs. For a basic SAR ADC, you need a sample-and-hold circuit, a reference voltage, and a comparator with a SAR to adjust the binary output.

812. Reading the ADC from your microcontroller returns some value. Given the value and the ADC supply voltages, convert this reading into a voltage.

Conversion: If the ADC value is 500 out of 1023 (10-bit ADC) and the supply voltage is 3.3V:

Voltage = (ADC Value / Max ADC Value) × Supply Voltage

Voltage = (500 / 1023) × 3.3V ≈ 1.61V

813. What is sampling?

Sampling: Taking periodic measurements of an analog signal to convert it into digital form.

814. What is sample and hold?

Sample and Hold: A circuit that takes a snapshot of the analog voltage at a specific time and holds it steady for the ADC to read.

815. How is the sample voltage maintained throughout the entire sample duration?

Maintaining Voltage: The sample-and-hold circuit uses a capacitor to store the voltage value during the sampling period.

816. Design a sample and hold circuit.

Design: Include a switch (like a MOSFET) to control sampling, a capacitor to hold the sampled voltage, and a buffer amplifier to stabilize the voltage for the ADC.

817. What are the tradeoffs between super fast/slow sample and hold durations?

Fast Duration: Less time to sample, which might cause errors if the signal changes quickly.

Slow Duration: More time to sample, which is better for stable signals but can reduce the overall speed of data acquisition.

818. What are some ADC frontends?

Frontends: Includes components like amplifiers to boost the signal before it enters the ADC, and filters to remove noise.

819. What is the precision of an ADC? Calculate it given all the required info.

Precision: Defined by the number of bits in the ADC. For example, a 12-bit ADC has 2^12 (4096) discrete values. Higher bits mean more precise conversion.

Precision = (Full Scale Voltage / 2^n)

DAC (Digital-to-Analog Converter)

Most of my ADC/DAC questions have been in the context of embedded systems, not too much on analog design or anything.

820. How does a DAC work?

DAC: Converts digital values into an analog signal. It takes a digital input and outputs a corresponding voltage or current.

821. Design a DAC.

Design: Use a resistor network or a digital-to-analog conversion IC. Ensure you have a stable reference voltage and appropriate output stage.

822. What is the precision of a DAC? Calculate it given all the required info.

Precision: Similar to ADCs, defined by the number of bits. For a 12-bit DAC, it can produce 2^12 (4096) discrete levels. Precision = (Full Scale Voltage / 2^n)

Data Storage and Memory

823. How does a microcontroller store memory and data?

Storage: Uses different types of memory like Flash (for program storage), RAM (for temporary data), and EEPROM (for non-volatile data).

824. If I need to store more memory than what the microcontroller has space for, what can I do?

Options: Use external memory chips like Flash or EEPROM connected via communication interfaces (e.g., SPI, I2C).

825. What if I want memory to persist after the microcontroller powers off?

Non-Volatile Memory: Use memory types like Flash, EEPROM, or FRAM that retain data without power.

826. What does non-volatile (NV) memory mean?

Non-Volatile Memory: Retains stored data even when power is lost.

827. What is ROM?

ROM: Stands for Read-Only Memory. It stores firmware or software that doesn't change and is not writable during normal operation.

828. What is an EEPROM?

EEPROM: Stands for Electrically Erasable Programmable Read-Only Memory. It allows for electrical rewriting and erasing of stored data.

829. What are registers?

Registers: Small, fast storage locations in a microcontroller used to hold data or instructions temporarily during processing.

830. Given a register map and the register value in hex, find the value of each register field.

Find Values: Convert the hex value to binary, then interpret each bit or group of bits based on the register map to determine the field values.

Debugging Embedded Systems

831. How do you debug a microcontroller? What are some challenges at hardware vs software levels?

Debugging: Use tools like debuggers and oscilloscopes. Challenges include understanding hardware issues (e.g., broken connections) versus software issues (e.g., coding errors).

832. What is a debugger?

Debugger: A tool that helps you test and debug code by allowing you to set breakpoints, step through code, and inspect variables.

833. What if your IDE/microcontroller doesn't have a hardware debugger?

Alternative: Use software debugging techniques like adding logging or using simple debugging tools like serial output for status messages.

834. If a microcontroller isn't booting up, what should you check for?

Checks: Ensure power supply is correct, reset pin is functioning, and that the firmware is correctly programmed and free of errors.

Operating Systems

835. What is an operating system?

Operating System (OS): Software that manages hardware and provides services for applications. It handles tasks like memory management, process scheduling, and input/output operations.

836. Name some operating systems.

Examples: Windows, Linux, macOS, Android, iOS, and RTOS (Real-Time Operating System) for embedded systems.

837. When is it needed?

Needed: When you require advanced features like multitasking, memory management, and user interfaces, or for managing complex hardware.

838. What is bare-metal programming?

Bare-Metal Programming: Writing software directly for the hardware without an operating system, giving you full control over the hardware but requiring more effort for managing tasks.

839. What advantages does it have over bare-metal?

RTOS Advantages: Provides multitasking, easier hardware management, and higher-level abstractions, simplifying complex applications and improving development efficiency.

840. What is a real-time operating system (RTOS)?

RTOS: An operating system designed to handle real-time applications where timing is critical. It ensures that high-priority tasks are completed within strict time constraints.

Sensors

841. What is a sensor?

Sensor: A device that detects physical changes (like temperature, pressure, or light) and converts them into electrical signals.

842. What are some applications of sensors?

Applications: Used in various fields such as automotive (for collision detection), healthcare (for monitoring vital signs), and consumer electronics (for touchscreens).

843. Name some types of sensors. What are they used for?

Types:

- Temperature Sensors: Measure temperature (e.g., thermocouples).
- Pressure Sensors: Measure pressure (e.g., barometers).
- Proximity Sensors: Detect the presence of objects (e.g., ultrasonic sensors).
- Light Sensors: Measure light intensity (e.g., photodiodes).

844. What is a transducer?

Transducer: A device that converts one form of energy into another. For example, a temperature sensor converts heat into an electrical signal.

845. What is sensor fusion?

Sensor Fusion: Combining data from multiple sensors to get more accurate or reliable information than using a single sensor alone.

Gyroscope/Accelerometer

846. What is a gyroscope/accelerometer?

A gyroscope measures angular velocity (how fast something is rotating).

An accelerometer measures linear acceleration (how fast something is moving in a straight line).

847. What are its applications?

Gyroscopes: Used in drones, smartphones, gaming controllers, and vehicle stability control.

Accelerometers: Used in step counters, vibration detection, and orientation sensing.

848. What is the difference between a gyroscope and accelerometer? Why are they often used together?

Gyroscope measures rotation, while the accelerometer measures movement in a straight line.

They are used together for more accurate motion detection, like in smartphones or drones, because the accelerometer gives movement data, and the gyroscope adds rotation information.

849. How can you determine position/velocity from the accelerometer data?

By integrating the acceleration data over time, you can calculate velocity. By integrating velocity, you can estimate position. However, this can lead to errors due to noise and drift.

850. What is an inertial measurement unit (IMU)? What are its tradeoffs?

An IMU is a sensor that combines a gyroscope and an accelerometer (and sometimes a magnetometer) to measure movement and orientation.

Tradeoffs: IMUs give a lot of data, but their readings can drift over time, leading to inaccurate measurements without corrections.

Light Measurement

851. What can you use to measure light? What does that even mean?

Devices like photoresistors and photodiodes measure light by detecting changes in light intensity. This helps understand how much light is present in an environment.

852. What properties of light do you look for? How can those properties be detected?

Properties like intensity, color, and direction. These can be detected using sensors like photodiodes (for intensity), color sensors, and light detection arrays.

853. What are the tradeoffs of photoresistors vs photodiodes?

Photoresistors: Simple and cheap but slow in response time.

Photodiodes: Faster and more accurate but more complex and expensive.

854. What are some applications of light measurement?

Ambient light sensing in phones, automatic lighting systems, solar panels, and photography.

855. How can light measurement be used in conjunction with other sensors to determine stuff?

Light sensors, combined with motion sensors or temperature sensors, can help automate systems like security lights that turn on when it’s dark and motion is detected.

Camera

856. How does a camera work?

A camera captures light through a lens and focuses it on a sensor (like a CCD or CMOS). The sensor converts the light into electrical signals to create an image.

857. What is optical image stabilization (OIS)? What is electronic image stabilization (EIS)? What are their differences and applications?

OIS: Physically adjusts the lens or sensor to counteract motion, used in cameras for clearer images in shaky conditions.

EIS: Uses software to stabilize the image by cropping and processing frames. It's common in smartphones and video stabilization.

OIS is more effective for still photography, while EIS is used in video.

858. How does night mode on smartphone cameras work?

Night mode takes multiple low-light images and combines them to produce a brighter, clearer photo by reducing noise and enhancing details.

859. What are some techniques to reduce image noise?

Techniques include using longer exposure times, taking multiple images and combining them, and using software algorithms to filter out noise.

860. How does ambient lighting impact camera performance?

Good lighting helps capture clearer and more detailed images, while poor lighting can cause grainy or blurry pictures.

861. How do you characterize camera quality?

Resolution, color accuracy, low-light performance, lens sharpness, and noise reduction are key factors.

862. Describe the camera processing pipeline.

Light enters through the lens → Hits the sensor → Converted to digital signals → Processed for color balance, sharpness, and noise reduction → Saved as an image or video.

863. How does a smartphone take pictures? Describe what happens when you press the button (and before).

Before pressing: The camera focuses and adjusts exposure and white balance.

When pressing: The sensor captures light, processes the data, and saves the image using software enhancements like noise reduction and sharpening.

LiDAR (Light Detection and Ranging)

864. What is LiDAR?

LiDAR uses lasers to measure distances by sending out light pulses and timing how long they take to bounce back.

865. How does it work?

A laser emits light → It hits an object → The reflected light is captured by a sensor → The time taken to return is used to calculate the distance to the object.

866. What are some applications?

Used in autonomous vehicles, drones, mapping, and 3D scanning.

867. What is the difference between LiDAR and a regular camera?

LiDAR measures distance and depth using lasers, while a camera captures visual images. LiDAR creates 3D maps, and cameras provide 2D images.

868. How do you map the world 360 degrees all around you?

Use multiple LiDAR sensors arranged to scan the surroundings or a rotating LiDAR sensor to capture data from all directions.

869. How would you describe the product of a LiDAR sensor?

A 3D point cloud, which is a detailed map of distances in the environment, showing object shapes and locations.

870. What are some factors that may impact accuracy?

Weather conditions (rain, fog), reflective surfaces, and sensor placement can affect the accuracy of LiDAR readings.

Calibration, Validation, and Testing

871. How do you calibrate/validate/test a sensor?

To calibrate a sensor, you compare its output with a known standard. For example, if you're testing a temperature sensor, you can compare its readings with a highly accurate thermometer.

Design an experiment: Use the sensor to measure a range of known values (e.g., different temperatures). Record the sensor's output and compare it to the reference. Adjust or correct the sensor’s readings if needed.

872. What are some things you want to validate in a sensor?

Accuracy (how close the readings are to true values)

Precision (how repeatable the readings are)

Response time (how quickly it reacts to changes)

Range (the span of values it can measure)

873. How do you characterize measurement error?

Measurement error is the difference between the sensor's reading and the true value. You can find this by testing the sensor against a known standard.

With this information, you can adjust the sensor (calibration) or compensate for errors in software.

Digital Basics

874. What is the difference between analog and digital?

Analog signals vary continuously (like a dimming light).

Digital signals have only two levels (like a light switch: on or off).

875. What is the difference between big and small endian?

Big endian: The most significant byte (MSB) is stored first.

Little endian: The least significant byte (LSB) is stored first.

876. What does MSB and LSB refer to?

MSB: Most Significant Bit (the highest value bit in a binary number).

LSB: Least Significant Bit (the lowest value bit).

Digital Electronics

877. How can you interface two devices with different logic levels?

You can use a level shifter to convert the voltage levels between two devices.

878. Design a level shifter.

A common level shifter uses MOSFETs or transistors to translate voltages between two logic systems (e.g., 5V to 3.3V).

879. Design a single-FET level shifter (passives are allowed).

Use a n-channel MOSFET. Connect the source to the lower voltage side, the drain to the higher voltage side, and add pull-up resistors to both sides.

880. What are setup and hold times?

Setup time: The minimum time the data must be stable before the clock edge.

Hold time: The minimum time the data must remain stable after the clock edge.

If violated, data corruption or timing errors occur.

881. What is the difference between a flip-flop and latch?

A latch is level-triggered (output changes when the input is stable).

A flip-flop is edge-triggered (output changes only on a clock edge).

882. How does a flip flop/latch work?

A flip-flop stores a bit of data and changes output on clock edges.

A latch changes output whenever the enable signal is active.

883. Design a memory cell.

Use a D flip-flop to store one bit of data. The data input is stored on the clock's rising edge, and the output holds until the next clock pulse.

884. Why are clocks needed?

Clocks synchronize operations in digital circuits, ensuring that all processes run at the same time.

885. What happens if a clock is too fast?

Data might not have enough time to stabilize, causing timing errors or metastability.

886. What is a register?

A register is a small amount of storage in a CPU that holds data temporarily.

887. What is metastability?

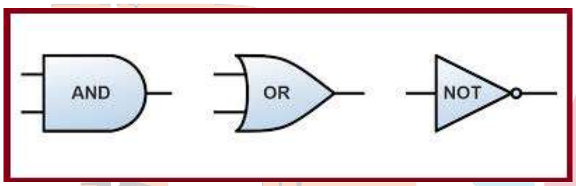
Metastability occurs when a system cannot decide on a stable output, often due to timing violations, causing unpredictable behavior.

888. What is a mux?

A multiplexer (mux) selects one of several input signals and forwards it to a single output.

Logic Gates

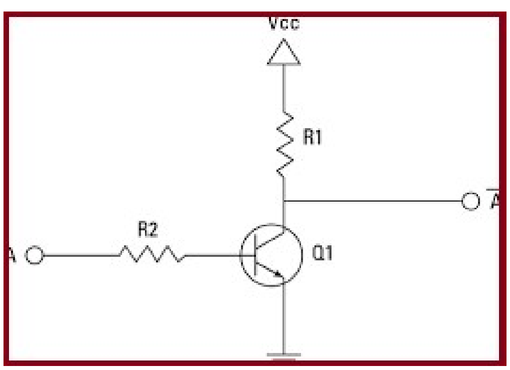
889. Draw a NOT/AND/etc. gate symbol.

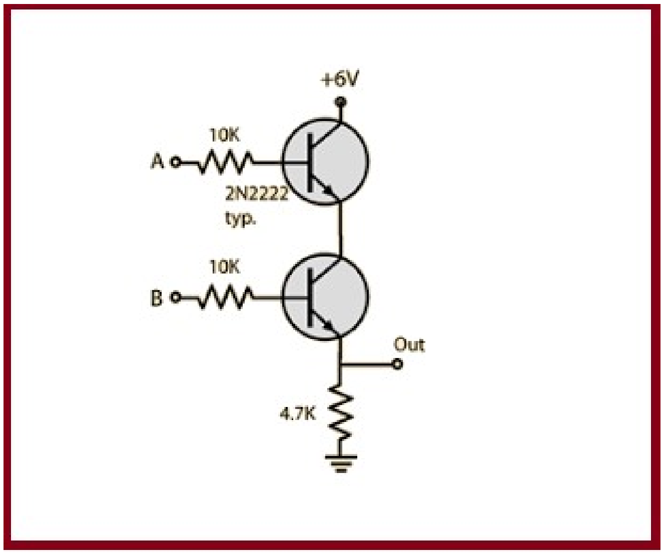


NOT gate: Triangle with a small circle at the output.

AND gate: A D-shaped symbol with two inputs on the left and one output on the right.

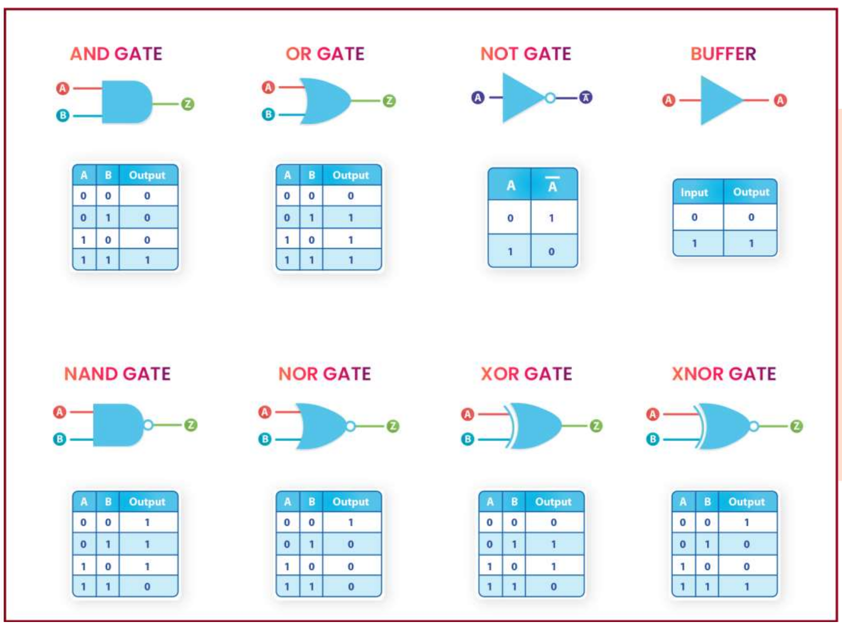
890. Draw a NOT/AND/etc. gate from transistors.





An AND gate can be built using two series-connected transistors.

891. Draw the truth table for NOT/AND/etc. gate.



892. What is the universal logic gate?

The NAND gate is universal, meaning you can create any other gate (AND, OR, NOT, etc.) using just NAND gates.

893. Convert some binary/hex/oct/decimal number to a different base.

Example: Binary 1010 = Decimal 10, Hex A, Octal 12.

Digital Logic

894. What is a truth table?

A truth table shows the output for every possible combination of inputs in a logic gate or circuit.

895. Do the K-map for a given truth table.

A Karnaugh map (K-map) simplifies Boolean algebra expressions by grouping terms to eliminate variables.

896. What is an FSM?

An FSM (Finite State Machine) is a computational model that moves between a set of states based on inputs.

897. What is the difference between a Mealy and Moore FSM?

Mealy FSM: Output depends on both the current state and the input.

Moore FSM: Output depends only on the current state.

VLSI/Computer Architecture

Computer Architecture

898. Name some computer architectures.

ARM, x86, RISC-V, MIPS, PowerPC.

899. What type of computer architectures are often used in smartphones/consumer electronics?

ARM architecture is commonly used due to its efficiency in power consumption.

900. What is a CPU?

The CPU (Central Processing Unit) is the brain of the computer that executes instructions and processes data.

901. What are some blocks in a CPU?

ALU (Arithmetic Logic Unit), Registers, Control Unit, and Cache.

902. What is an ALU?

The ALU performs arithmetic and logical operations (e.g., addition, subtraction, AND, OR).

903. What is the program counter?

The Program Counter (PC) holds the memory address of the next instruction to be executed.

904. How does the contents of the program counter change over time?

It typically increments after each instruction is fetched or can jump to a new address in the case of a branch or function call.

905. What is the fetch/execute cycle?

The fetch/execute cycle involves:

Fetch: Getting the next instruction from memory.

Decode: Understanding the instruction.

Execute: Performing the instruction.

906. Describe the five-stage pipeline.

Fetch: Get instruction.

Decode: Interpret instruction.

Execute: Perform operation.

Memory access: Read/write data.

Write-back: Store results.

907. How can you offload tasks from the CPU?

By using specialized hardware (e.g., GPU, DSP, or co-processors) for parallel tasks, which can speed up performance but adds complexity and may consume more power.

908. What can you do to increase the throughput of a CPU?

Pipelining, increasing clock speed, adding more cores, or improving branch prediction.

909. What is pipelining?

Pipelining allows multiple instructions to be processed simultaneously at different stages, increasing CPU efficiency.

910. What is branch prediction?

Branch prediction guesses the outcome of a decision in a program to keep the pipeline running smoothly.

Algorithms include simple guesses (always taken) or more complex ones (history-based prediction).

911. What are interrupts?

Interrupts temporarily stop the CPU to handle an important event (e.g., input from a keyboard). They are critical for real-time responses, especially in embedded systems.

912. What is microarchitecture?

Microarchitecture is the internal design of a CPU (how components like ALU, cache, etc., are organized).

913. What is an instruction set architecture (ISA)?

The ISA defines the set of instructions a CPU can execute (e.g., ARM, x86).

914. What is the difference between microarchitecture and ISA?

ISA is the instructions a CPU understands, while microarchitecture is how the CPU is built to execute those instructions.

915. What is RISC-V?

RISC-V is an open-source, reduced instruction set architecture (RISC).

916. Does RISC-V specify the ISA or microarchitecture?

RISC-V specifies the ISA, not the microarchitecture.

917. What are some advantages of RISC vs CISC?

RISC (Reduced Instruction Set Computing) is simpler, more efficient, and uses fewer instructions, while CISC (Complex Instruction Set Computing) has more complex instructions but may need fewer cycles per instruction.

918. How do you characterize CPU performance?

Clock speed, number of cores, instructions per cycle, and cache size are used to measure CPU performance.

GPU

919. What is a GPU?

A GPU (Graphics Processing Unit) is specialized hardware designed for rendering images and performing parallel processing tasks.

920. How does a GPU work?

A GPU processes many operations in parallel, which makes it well-suited for handling graphical tasks like rendering images or video.

921. What does the GPU pipeline look like?

The GPU pipeline includes stages like vertex processing, rasterization, and fragment processing, which are used to render 3D graphics.

922. What is the difference between a GPU and CPU?

A CPU is designed for general-purpose tasks and single-thread performance, while a GPU excels in parallel tasks like graphics rendering.

923. What optimizations does a GPU have so it vastly outperforms CPUs in graphics tasks?

A GPU has many parallel cores optimized for tasks that can be broken into smaller jobs, but it’s less flexible for general tasks.

924. How do you characterize GPU performance?

Clock speed, number of cores, memory bandwidth, and FLOPS (Floating Point Operations Per Second).

925. What are other tasks that GPUs may excel at besides graphics processing?

Machine learning, cryptocurrency mining, and scientific simulations due to their parallel processing power.

System on Chip (SoC)

926. What is a SoC?

A System on Chip (SoC) integrates multiple components like CPU, GPU, memory, and other peripherals onto a single chip.

927. What are its applications?

SoCs are used in smartphones, IoT devices, and embedded systems where space and power efficiency are critical.

928. What are some components on a SoC?

CPU, GPU, RAM, I/O controllers, network interfaces.

929. What are some tradeoffs between going for a SoC vs all separate components?

SoC is compact and power-efficient, but harder to upgrade or customize. Separate components offer flexibility but take up more space and use more power.

930. When designing a system, what other components are needed in addition to a SoC?

Power management, storage (e.g., flash memory), and connectors for external devices (e.g., USB, sensors).

931. What is a system on module (SoM)? What is the difference/tradeoffs vs a SoC?

An SoM is a pre-built circuit board containing an SoC, memory, and other components. It simplifies integration but may be larger than a custom SoC solution.

932. What are some tradeoffs of integration?

More integration leads to space and power savings, but makes repair and upgrading harder.

933. Which blocks of the SoC consume the most power?

The CPU, GPU, and wireless communication modules typically consume the most power.

Memory

934. Where can data and programs be stored in memory?

Data and programs can be stored in RAM (Random Access Memory), ROM (Read-Only Memory), and storage devices like flash memory or hard drives.

935. What is the difference between data and program memory?

Program memory stores the instructions the CPU executes (code), while data memory stores the variables and data the program uses.

936. Explain the memory hierarchy. What are the tradeoffs of each?

Memory hierarchy includes:

Registers (fastest, smallest size, used by CPU directly).

Cache (fast, close to CPU, stores frequently used data).

RAM (slower, larger, temporary data storage).

Storage (slowest, permanent storage like SSDs or HDDs).

Tradeoffs: Faster memory is smaller and more expensive; slower memory is larger and cheaper.

937. What is a cache?

A cache is a small, fast memory that stores frequently accessed data to speed up processing.

938. What are registers?

Registers are very small, fast memory units inside the CPU that store data for immediate use by the processor.

939. What is RAM?

RAM (Random Access Memory) is temporary storage used by programs while they run. It is cleared when the device is powered off.

940. What does "random access" refer to?

Random access means data can be read from or written to any location in memory at any time, without having to go through other data first.

941. What is stack overflow?

A stack overflow occurs when too much data is placed on the stack, exceeding its allocated memory, causing errors or crashes.

942. What is the difference between the stack and the heap?

The stack is for managing function calls and local variables, while the heap is for dynamically allocated memory that you can control (e.g., for large or unpredictable data structures).

943. What is a memory management unit (MMU)?

The MMU manages and translates memory addresses between the CPU and physical memory, allowing processes to use virtual memory for protection and efficiency.

944. If I'm operating my computer in space where there can be lots of radiation that can corrupt my memory, what issues can arise?

Radiation can cause bit flips in memory, leading to incorrect data or crashes. Solutions include using error-correcting codes (ECC) and radiation-hardened hardware.

Power, Performance, and Thermals

945. How does clock frequency impact power, performance, and thermals?

Higher clock frequency increases performance but also raises power consumption and generates more heat.

946. Why are multiple clock domains needed?

Different parts of a system may run at different speeds to optimize power efficiency and performance.

947. How does supply voltage impact power, performance, and thermals?

Higher voltage increases performance but also increases power consumption and heat. Lowering voltage can save power but may reduce speed.

948. Why are multiple supply voltages needed?

Different components in a system may have different voltage requirements for power efficiency and performance optimization.

949. How does process, voltage, and temperature (PVT) impact power, performance, and thermals?

Process variations affect how a chip is made, voltage influences speed and power, and temperature affects how efficiently a chip runs. All three together determine overall performance and power usage.

950. What are process corners?

Process corners refer to different manufacturing scenarios (e.g., fastest or slowest chips) that affect a chip's performance and power consumption.

951. What are typical temperature ranges for validation?

Typically between -40°C to +85°C for consumer electronics, and up to +125°C for automotive or industrial applications.

952. What can you do to decrease power consumption?

Lower clock frequency, reduce supply voltage, use power-saving modes, and optimize software algorithms.

953. What can you do to throttle performance?

You can lower the clock speed, reduce core usage, or decrease the supply voltage.

954. What sort of temperature control loops are commonly implemented?

Thermal throttling, fan control, and dynamic voltage and frequency scaling (DVFS) are common loops to manage temperature.

955. What is power, performance, area (PPA)? Describe the tradeoffs.

PPA refers to the trade-offs between power consumption, performance, and area (size of the chip). Improving one often impacts the others (e.g., higher performance may need more power or a larger area).

RTL (Register Transfer Level)

956. Describe any experience you have with RTL.

RTL is used to describe the digital circuits and logic design at the register level using languages like Verilog or VHDL.

957. Design a clock divider.

A clock divider reduces the frequency of a clock signal, typically done by counting pulses and generating an output signal that toggles at the desired rate.

958. Design an arbiter. What are the different types and their tradeoffs?

An arbiter manages access to shared resources. Types include round-robin, priority-based, and FIFO. Tradeoffs involve fairness vs. simplicity and implementation complexity.

959. Design a mux.

A multiplexer (mux) selects one of several input signals and forwards it to the output based on control signals.

960. What is a module?

A module is a block of reusable code in RTL that can represent a functional unit, such as an adder or a register file.

961. How is RTL reused? Why is this important?
RTL is reused by modularizing design blocks, allowing faster development and verification. It reduces design time and errors, and helps in standardizing components for different designs.

Assembly

962. What is assembly? How does it differ from programming languages like C or Python?

Assembly is a low-level programming language that communicates directly with the hardware. Each instruction in assembly corresponds to a specific machine code operation. In contrast, languages like C and Python are high-level, meaning they abstract away hardware details, making them easier to read and write but slower to execute.

963. Describe what this given program is doing. How long will it take to run? How many times will {some subroutine} get executed?

To answer this, you would need to analyze the program step by step, look at its logic, and count the number of iterations or instructions. Time to run depends on factors like the number of instructions and processor speed.

Coding Experience

964. What sort of coding experience do you have?

Here, you would describe the coding experience you've gained through work or personal projects, including specific languages, tools, and environments you've worked in.

965. Do you have coding experience in:

C: Yes, embedded systems often use C for low-level hardware control.

C++: It is an extension of C, used in systems requiring object-oriented programming.

Python: A high-level scripting language, often used in automation, data processing, and test scripting.

966. What are your strongest coding languages? How would you rate yourself in them?

You would name your strongest languages (e.g., C, Python, C++) and rate your proficiency, typically on a scale from beginner to expert.

967. What is regex?

Regex (regular expression) is a tool used to find patterns in text. It's helpful for searching, matching, or manipulating strings based on defined patterns.

Variables

968. What are the basic variable types?

Common types include:

- int (integer)
- float (decimal numbers)
- char (character)
- bool (true/false)
- string (text or collection of characters)

969. What is the difference between strings and chars? How are strings usually implemented?

A char is a single character (e.g., 'A'), while a string is a sequence of characters (e.g., "Hello"). Strings are usually implemented as an array of characters with a null terminator ('\0') at the end.

970. What does the volatile keyword refer to? When should volatile variables be used?

volatile tells the compiler not to optimize the variable because its value may change unexpectedly, such as in hardware registers or during interrupts. Use volatile when a variable can change outside the program's control.

Programming Questions

971. Write pseudocode to achieve some given functionality.

Pseudocode is a plain English description of how an algorithm works. It's not real code but helps outline the logic before actual coding.

972. Walk through how a given program works.

Analyze the program step by step, explaining how each part works, from variable initialization to loops and function calls.

973. Write code to:

- Reverse a string
- Parse a file to find and/or modify specific text.
- Psuedocode API design.
- Find a substring in a string.
- Check if a string is a palindrome.

System Architecture

"System architecture" is one of the vaguest engineering words with no meaning. I literally but it on my resume for that reason LOL. These questions are mostly focused on systems at a higher level to achieve some given functionality.

974. What components does a smartphone/laptop have?

Smartphone/Laptop components typically include:

Processor (CPU/GPU): Handles computing tasks.

Memory (RAM/ROM): Stores data and instructions.

Storage: SSD or flash memory for long-term storage.

Battery: Power supply.

Display: Touchscreen or monitor.

Camera: For photos and video.

Sensors: Gyroscope, accelerometer, ambient light, etc.

Networking: Wi-Fi, Bluetooth, LTE/5G, Ethernet (laptop).

I/O ports: USB, HDMI, etc.

Speakers/Microphone: For audio input and output.

975. Design a system to achieve:

Variable LED brightness: Use a PWM (Pulse Width Modulation) signal to adjust the brightness by varying the duty cycle.

LED brightness based on ambient brightness: Use an ambient light sensor to read brightness levels and adjust the LED brightness accordingly via a control loop.

A security system: Use sensors (motion, door/window sensors), cameras, and controllers. Central control through a microcontroller or smart hub connected to a cloud for remote monitoring.

A camera system: Include a lens, image sensor, processor, and storage for capturing and processing images.

A tracking system: Use GPS or RFID with a microcontroller to track and report location data.

Eye tracking glasses: Incorporate infrared sensors or cameras to monitor eye movement, combined with a processor to interpret gaze direction.

976. What is redundancy in a system? When is it preferred? How can you design a system with redundancy?

Redundancy means having multiple components that perform the same function, ensuring the system remains operational if one fails. It is preferred in mission-critical systems (e.g., aerospace, medical devices) where failure can cause serious harm. Design can include backup components, failover mechanisms, or mirroring for data storage.

977. What are some system design constraints in consumer electronics vs automotive vs aerospace/space/defense vs medical devices?

Consumer electronics: Focus on cost, size, and battery life.

Automotive: Reliability, environmental durability, safety standards.

Aerospace/space/defense: Extreme reliability, radiation resistance, low power, and temperature endurance.

Medical devices: Safety, accuracy, regulatory compliance (e.g., FDA), and reliability.

Measurement

978. What tools can you use to measure stuff?

DMM (Digital Multimeter), Oscilloscope, VNA (Vector Network Analyzer), Power Supplies, SMU (Source Measure Unit).

979. How familiar are you with:

DMMs: Measure voltage, current, resistance, and sometimes frequency.

Scopes: Measure voltage over time for signal analysis.

VNAs: Measure network parameters like impedance and phase.

Power supplies: Provide stable power, sometimes with multiple channels or quadrants for advanced testing.

980. What sample rate/bandwidth will I need to measure a signal?

The sample rate should be at least 10 times the highest frequency of the signal (Nyquist rate). The bandwidth should also exceed the signal frequency to capture high-frequency components accurately.

981. How should I measure X signal?

Depends on the signal type. For low-frequency signals, use a DMM or scope. For high-frequency signals, use an oscilloscope with proper bandwidth.

982. How can I measure current?

Use a current probe or measure the voltage across a shunt resistor and apply Ohm's law.

983. What is the difference between bandwidth and sampling rate?

Bandwidth is the range of frequencies a tool can accurately measure. Sampling rate is how often a signal is captured per second.

984. What is a VNA?

A Vector Network Analyzer is a tool for measuring complex electrical networks' frequency response, typically used for RF testing.

985. How do I measure resistance/capacitance/inductance/impedance?

Use a DMM for resistance. For capacitance/inductance, specialized meters or a LCR meter are used. For impedance, a VNA or LCR meter can be employed.

DMMs (Digital Multimeters)

986. What is a DMM?

A Digital Multimeter measures voltage, current, resistance, and sometimes other quantities like frequency or temperature.

987. What can it measure?

Voltage, current, resistance, continuity, and in some models, capacitance or frequency.

988. What is a DMM's sample rate? Does it matter?

The sample rate is how often it takes measurements. It can matter when measuring fast-changing signals.

989. When to use DMM vs scope?

Use a DMM for slow, steady signals (like DC voltage). Use a scope for faster, more complex signals (like AC or waveforms).

Scopes

990. What is an oscilloscope?

A tool that visually displays electrical signals, showing how voltage changes over time.

991. How does an oscilloscope work?

It samples an electrical signal at regular intervals and displays it as a waveform on the screen.

992. What are triggers on a scope? How do I use them?

A trigger tells the oscilloscope when to start recording the signal. Use it to stabilize and synchronize the display to recurring events (like a clock signal).

993. How do I get current if I'm measuring the voltage across a sense resistor?

Apply Ohm's Law: Current = Voltage / Resistance.

994. When would I need multiple scope channels?

For analyzing multiple signals at once, such as comparing a signal at different points in a circuit.

Scope Probes

995. What is a diff probe? When do I need it?

A differential probe measures the voltage difference between two points without referencing ground. Useful when working with high-voltage or floating signals.

996. What is an active probe? When do I need it?

An active probe amplifies the signal and is used for high-speed or low-amplitude signals where a passive probe might not be accurate.

997. What does the 10x attenuation do on a probe?

It reduces the signal by a factor of 10 to allow for measurement of higher voltage signals without damaging the scope.

998. What is a current probe?

A current probe measures the current flowing through a conductor without breaking the circuit.

999. Probe input impedance - what should it be and why?

High input impedance is preferred (typically 10 MΩ) to avoid loading the circuit and altering the signal you're measuring.

1000. When would I want different probe impedance?

Use lower impedance when working with high-frequency signals to avoid signal reflection and distortion. Use higher impedance for low-frequency signals to avoid loading

General Debug, Testing, and Validation

This section is just for general debug and testing, some sections may have their own debug, testing, validation, and measurement subsections.

1001. What tools can you use to debug/test/validate stuff?

Multimeter (DMM) for basic electrical measurements (voltage, current, resistance).

Oscilloscope for analyzing time-varying signals.

Logic analyzer for debugging digital communication protocols.

Power supply for providing controlled power to devices.

Debugger (e.g., JTAG) for step-by-step software debugging.

Protocol analyzers for specific interfaces like USB, I2C, or SPI.

Software testing frameworks for validating software.

1002. What is your approach to debug/test/validate something?

Identify the problem: Gather details and understand what’s wrong.

Isolate the issue: Narrow down to the faulty component or software block.

Test systematically: Use tools (multimeter, scope, debugger) to check each part.

Monitor outputs: Compare expected vs. actual results.

Fix and retest: Apply solutions and ensure the problem is resolved.

1003. Design a system to test X.

Understand the test objective: What are you testing? (e.g., functionality, power, performance).

Set up test conditions: Define inputs, outputs, and environmental conditions.

Choose tools: Select hardware (DMM, scope) or software frameworks for testing.

Automate tests if needed: Use scripts to run tests multiple times.

Collect data: Record and analyze results to validate performance.

1004. How do you debug software?

Reproduce the bug: Try to recreate the issue consistently.

Use a debugger: Step through the code to find where things go wrong.

Check logs/print statements: Use logs to trace the flow of execution and spot errors.

Isolate the problem: Narrow down to the function or module where the bug occurs.

Fix and test: Apply the fix and rerun the software to confirm it's working correctly.

1005. Often times you need to run tests for days, weeks, or months continuously. How do you ensure these tests run smoothly 24/7 for the entire time? What can you do to protect against power outages?

Automation: Use automated scripts to manage continuous testing and monitoring.

Error handling: Implement mechanisms to detect and handle errors without stopping the test.

Backup power: Use uninterruptible power supplies (UPS) to protect against power outages.

Data logging: Ensure test results are logged and saved frequently to avoid data loss in case of interruptions.

Remote monitoring: Set up systems for remote test management and alerts for failure detection.

Non-Technical

A lot of these non-technical questions are the generic interview questions but they still do come up.

Intro

1006. How are you doing?

I'm doing well, thanks! (lmao).

1007. Introduce yourself.

I'm an engineering manager with 9 years of experience in embedded systems, robotics, and wireless technologies like BLE and IoT. I enjoy working on innovative projects, particularly in automation and embedded development.

1008. What are your interests/goals?

My interests include embedded systems, robotics, and IoT. My goal is to lead teams in developing advanced embedded technologies and smart devices.

1009. What are your career goals?

I aim to continue growing in embedded systems, leading innovative projects, and eventually manage larger engineering teams or work on cutting-edge technology products.

1010. What are you looking for in this position/company?

I'm looking for opportunities to apply my technical skills in embedded systems and contribute to impactful projects while also growing professionally.

1011. Why are you interested in this position/company?

I'm interested because the company is known for innovation, and the position aligns with my expertise in embedded systems and IoT.

1012. Do you have other competing interviews/offers/deadlines?

Yes, I'm considering other opportunities but am particularly interested in this one.

1013. When are you available for an internship?

(If applicable) I am available [insert timeframe].

1014. Do you require visa assistance?

(If applicable) Yes/No.

1015. Why don't you want to return to your previous internship/company?

(If applicable) I'm looking for new challenges and growth opportunities that align better with my current goals.

1016. What is your desired salary?

I am open to discussing a competitive salary based on the role and responsibilities.

1017. Are you okay with moving to X for an internship?

(If applicable) Yes, I'm flexible with relocation if it's the right opportunity.

Behavioral:

1018. What are your strengths/weaknesses?

Strengths: Leadership, technical expertise, problem-solving.

Weaknesses: I tend to take on too many tasks at once but am working on delegating more effectively.

1019. How would your friends and peers describe you?

Hardworking, dependable, and a good problem solver.

1020. What is your approach to solving hard problems?

Break the problem into smaller parts, analyze each, and systematically test potential solutions.

1021. How do you work under pressure?

I stay calm, prioritize tasks, and focus on resolving the most critical issues first.

1022. How do you work in groups?

I collaborate well, share ideas, listen to others, and ensure everyone's input is considered.

1023. Tell me about a time where you and a peer/coworker disagreed on something. How did you guys reconcile it?

We openly discussed our viewpoints, weighed the pros and cons of each, and ultimately came to a compromise that worked for both of us.

1024. Tell me about a time where you needed to juggle many tasks. How did you manage it?

I prioritized tasks based on deadlines and importance, delegated when necessary, and stayed organized with a clear schedule.

1025. What is the perfect job for you?

A role that challenges me technically, allows for creativity, and involves working on impactful, innovative products.

1026. What do you find important in a company?

A culture of innovation, growth opportunities, and a collaborative work environment.

1027. How much do you know about this company/this product?

I've researched the company and its products and admire how it leads in [mention specific area]. I'm excited to learn more.

1028. Why should I hire you?

I bring strong technical skills, proven leadership, and a track record of successfully delivering projects in embedded systems and robotics.

1029. How do you objectively assess standards of success? How do you meet these?

I set clear, measurable goals and regularly review progress to ensure I'm on track, adjusting as needed to stay aligned with the objectives.

1030. Describe a time where change became a problem. How did you overcome it?

During a project, requirements changed mid-way. I adapted by reworking the design and closely communicating with stakeholders to ensure the new direction was feasible.

1031. Describe a time where you encouraged different perspectives in a group setting.

I facilitated a brainstorming session, ensuring everyone had a chance to share their ideas, which helped us find creative solutions.

1032. How do you balance school with extracurricular activities?

(If applicable) I manage my time carefully, setting clear priorities and making time for both academics and activities.

1033. What are your top three skills that contribute to you being a good employee?

Technical expertise, leadership, and communication.

1034. Describe a time of new opportunity for improvement.

After reviewing past projects, I identified areas where automating certain processes could save time and improve efficiency, leading to significant workflow improvements.

1035. Describe a time where you didn't meet expectations. What happened?

I underestimated the complexity of a task, causing a delay. I learned to allocate more buffer time and plan for unexpected challenges.

1036. How have you taken advantage of diversity (like at school)?

I've worked with diverse teams, learning different perspectives and approaches, which has helped me broaden my problem-solving abilities.

1037. Describe a time you admitted to a mistake.

I once missed a crucial detail in a design. I quickly admitted the mistake, took responsibility, and worked with the team to correct it before it became a bigger issue

Technical/Behavioral

Behavioral questions that are more on the technical side or can get more technical, but not technical enough to be placed with the other technical questions.

1038. Tell me about a project you're proud of.

I worked on a project where I helped design a simple app that tracks daily habits. It was satisfying to see how a tool I built could help people stay organized and motivated.

1039. Tell me about a time you debugged something/solved a difficult problem.

Once, my code for a game wasn't working as expected. After checking the error messages and the logic, I realized I had a small mistake in the code that caused the problem. Fixing that mistake made the game run smoothly.

1040. What is your approach to debug something?

First, I try to understand what the problem is and when it happens. Then, I check if there are any error messages or clues. I'll break the problem into smaller parts and test each part to find where it's going wrong. Finally, I fix the issue and test again to make sure it's solved.

1041. If you need to make a presentation about x, how would you present the data?

I'd start with a clear title and introduce the topic. Then, I'd use charts or graphs to show the important data. I'd explain the main points simply and use examples to make it easier to understand. I'd finish with a summary of key takeaways.

1042. What coursework have you taken?

I've taken classes in programming, data analysis, and software development.

1043. How familiar are you with:

Programming: I know how to write code in languages like Python and JavaScript.

Some specific programming language: I'm familiar with Python, which I use for various tasks like data analysis and automation.

Some specific software: I've used software like Microsoft Excel for data handling and Git for version control.

Some technical role/function: I've done some work in coding and testing software to make sure it works as intended.

Some technical subject: I understand basic concepts in computer science, like algorithms and data structures.

Some specific tools: I've used tools like multimeters for basic electrical measurements.

Some specific measurement device: I've used Digital Multimeters (DMMs) to measure voltage, current, and resistance.

1044. What sort of coding experience do you have?

I have experience coding in several languages and building small projects like websites and simple applications.

1045. If you had to give a 20-minute TED Talk, what subject would you choose?

I'd choose to talk about the impact of technology on everyday life and how it can be used to solve common problems.

1046. What is a datasheet?

A datasheet is a document that provides detailed information about a product or component, like its specifications, features, and how to use it.

1047. How do you read it?

To read a datasheet, start with the overview to understand what the component does. Then look at the specifications to see its technical details and limits. Check the diagrams or examples to see how it should be used.

1048. Given this datasheet, find some spec or how something works.

To find specific information, look at sections like "Specifications" for performance details or "Pin Configuration" for how to connect the component. Use these details to understand how the component fits into your project.

And that's the list! I hope you found it helpful! If you have any questions, please leave a public comment below! I really love it when people ask questions publicly because then **everyone** can learn from it so please keep that up! I will likely be updating this list as I get more questions/rescan my interview notebooks (At the time of writing this for the first time, this list has 1048 questions).

Appendix

- AOI - Automated Optical Inspection
- BGA - Ball Grid Array
- BLDC - Brushless DC Motor
- C - Capacitor/Capacitance or the programming language
- CSA - Current Sense Amplifier
- DFM - Design for Manufacturing
- DFT - Design for Testing
- DFX - Design for Excellence
- DRC - Design Rule Checking
- ECAD - Electrical Computer-Aided Design
- EIS - Electronic Image Stabilization
- EMI - Electromagnetic Interference
- EMR - Electromechanical Relay
- ESC - Electronic Speed Controller
- FCT - Functional Testing
- FTE/FT - Full-Time Engineer/Full-Time
- ICT - In-Circuit Testing
- IGBT - Insulated-Gate Bipolar Transistor
- IMU - Inertial Measurement Unit
- ISA - Instruction Set Architecture
- ISR - Interrupt Service Routing
- L - Inductor/Inductance
- LDO - Low-Dropout Regulator
- LiPo - Lithium Polymer
- MMU - Memory Management Unit
- NV - Non-Volatile
- OIS - Optical Image Stabilization
- OOP - Object-Oriented Programming
- PoP - Package on Package
- PCB - Printed Circuit Board
- PI - Power Integrity
- PPA - Power, Performance, Area
- PVT - Process/Voltage/Temperature
- R - Resistor/Resistance
- RTOS - Real-Time Operating System
- SI - Signal Integrity
- SI/PI - Signal Integrity/Power Integrity
- SoC - State of Charge (for a battery) or System on Chip
- SoM - System on Module
- SSR - Solid-State Relay
- TL - Transmission Line