ELTR 140 (Digital 1), section 3

Recommended schedule

Day 1

    Topics: *Numeration systems*
    Questions: *1 through 10*
    Lab Exercise: *work on project*

Day 2

    Topics: *Digital codes*
    Questions: *11 through 20*
    Lab Exercise: *Gray code to binary converter (question 56)*

Day 3

    Topics: *Binary arithmetic*
    Questions: *21 through 30*
    Lab Exercise: *Half adder circuit (question 57)*

Day 4

    Topics: *Binary arithmetic circuits*
    Questions: *31 through 40*
    Lab Exercise: *Full adder circuit (question 58)*

Day 5

    Topics: *Digital circuit troubleshooting*
    Questions: *41 through 55*
    Lab Exercise: *Analog-digital converter IC (question 59)*

Day 6

    Exam 3: *includes binary adder circuit performance assessment*
    Project due
    Question 60: Sample project grading criteria

DC/AC review problems

    Questions: *61 through 80*

General concept practice and challenge problems

    Questions: *81 through the end of the worksheet*

Skill standards addressed by this course section

F Technical Skills – Digital Circuits

F.05 Understand principles and operations of types of logic gates.

F.20 Understand principles and operations of types of arithmetic-logic circuits.

B Basic and Practical Skills – Communicating on the Job

B.01 Use effective written and other communication skills. *Met by group discussion and completion of labwork.*

B.03 Employ appropriate skills for gathering and retaining information. *Met by research and preparation prior to group discussion.*

B.04 Interpret written, graphic, and oral instructions. *Met by completion of labwork.*

B.06 Use language appropriate to the situation. *Met by group discussion and in explaining completed labwork.*

B.07 Participate in meetings in a positive and constructive manner. *Met by group discussion.*

B.08 Use job-related terminology. *Met by group discussion and in explaining completed labwork.*

B.10 Document work projects, procedures, tests, and equipment failures. *Met by project construction and/or troubleshooting assessments.*

C Basic and Practical Skills – Solving Problems and Critical Thinking

C.01 Identify the problem. *Met by research and preparation prior to group discussion.*

C.03 Identify available solutions and their impact including evaluating credibility of information, and locating information. *Met by research and preparation prior to group discussion.*

C.07 Organize personal workloads. *Met by daily labwork, preparatory research, and project management.*

C.08 Participate in brainstorming sessions to generate new ideas and solve problems. *Met by group discussion.*

D Basic and Practical Skills – Reading

D.01 Read and apply various sources of technical information (e.g. manufacturer literature, codes, and regulations). *Met by research and preparation prior to group discussion.*

E Basic and Practical Skills – Proficiency in Mathematics

E.01 Determine if a solution is reasonable.

E.02 Demonstrate ability to use a simple electronic calculator.

E.06 Translate written and/or verbal statements into mathematical expressions.

E.12 Interpret and use tables, charts, maps, and/or graphs.

E.13 Identify patterns, note trends, and/or draw conclusions from tables, charts, maps, and/or graphs.

E.15 Simplify and solve algebraic expressions and formulas.

E.16 Select and use formulas appropriately.

Common areas of confusion for students

Difficult concept: *Purpose of Gray code on encoder wheels.*

The purpose of using Gray code instead of binary encoding on rotary encoder wheels is a difficult concept for some to grasp. One aid I have found in explaining the need for Gray encoding is to build an actual rotary encoder wheel out of a piece of stiff card-stock paper, drawing the code bits on the wheel as semi-circular arcs. By rotating this wheel and looking at the bits go by, you can better see what problems may result if the bit sensors were out of alignment.

Difficult concept: *Arithmetic overflow.*

When we represent numbers in binary (or any other numeration system, for that matter!) given a fixed number of characters, we become bound to a certain range of representable numbers. A 10-digit electronic calculator lacking powers-of-ten notation, for example, can only display numbers from -9,999,999,999 to +9,999,999,999. Any attempt to calculate answers beyond this range will result in *overflow*, as the digital circuit will not be able to represent the result. It is important for students to comprehend and respect the concept of overflow, because it allows intelligent interpretation of digital calculation results. It is necessary for a digital circuit to be able to detect this condition, so that the answer may be flagged as incorrect, rather than have the answer be presented as valid.

Questions

Digital computers use a numeration system with a base of *two*, rather than a base of *ten* as we are accustomed to using. It is much easier to engineer circuitry that counts in "binary" than it is to design circuits that count in any other base system. Based on what you know of numeration systems, answer the following questions:

- How many different symbols (ciphers) are there in the binary numeration system?
- What are the different place-weight values in the binary system?
- How would you represent the number "seventeen" in binary? 10001
- In our base-ten (denary) numeration system, each character is called a "digit." What is each character called in the binary numeration system? Bit

file 01198

Question 2

What is the largest number that can be counted to, in a base-ten system with six digits? How about in a base-twenty (vigesimal) system with four places? How about in a base-two (binary) system with ten bits?

file 01200   999,999
                                        JJJJ

Question 3

Convert the following numbers from binary (base-two) to decimal (base-ten):

- $10_2 = $ 2
- $1010_2 = $ 10
- $10011_2 = $ 19
- $11100_2 = $ 28
- $10111_2 = $ 23
- $101011_2 = $ 43
- $11100110_2 = $ 230
- $10001101011_2 = $ 1131

Describe a general, step-by-step procedure for converting binary numbers into decimal numbers.

file 01202

Question 4

Convert the following numbers from decimal (base-ten) to binary (base-two):

- $7_{10} = $ 111
- $10_{10} = $ 1010
- $19_{10} = $ 10010
- $250_{10} = $ 11111001
- $511_{10} = $ 111111111
- $824_{10} = $ 1100111000
- $1044_{10} = $ 10000010100
- $9241_{10} = $ 10010000011001

Describe a general, step-by-step procedure for converting decimal numbers into binary numbers.

file 01204

Question 5

A numeration system often used as a "shorthand" way of writing large binary numbers is the *octal*, or base-eight, system.

Based on what you know of place-weighted numeration systems, describe how many valid ciphers exist in the octal system, and the respective "weights" of each place in an octal number.

Also, perform the following conversions:

1,8,64,512,4096

- 35_8 into decimal: 29
- 16_{10} into octal: 20
- 110010_2 into octal:
- 51_8 into binary: 101001

file 01286

Question 6

A numeration system often used as a "shorthand" way of writing large binary numbers is the *hexadecimal*, or base-sixteen, system.

Based on what you know of place-weighted numeration systems, describe how many valid ciphers exist in the hexadecimal system, and the respective "weights" of each place in a hexadecimal number.

Also, perform the following conversions:

1,16,256,4096

- 35_{16} into decimal: 53
- 34_{10} into hexadecimal: 22
- 11100010_2 into hexadecimal: E2
- 93_{16} into binary: 10010011

file 01206

Question 7

Complete this table, performing all necessary conversions between numeration systems:

Binary	Octal	Decimal	Hexadecimal
10010	22	18	12
1011100	134	92	5C
101010	52	42	1A
110111	67	55	37
1100101	145	101	65
100100010	442	290	122
1111101000	1750	1000	3E8
11011110	336	222	DE
1011010110	1326	726	2D6

file 01207

Question 8

Digital computers communicate with external devices through *ports*: sets of terminals usually arranged in groups of 4, 8, 16, or more (4 bits = 1 *nybble*, 8 bits = 1 *byte*, 16 bits = 2 bytes). These terminals may be set to high or low logic states by writing a program for the computer that sends a numerical value to the port. For example, here is an illustration of a microcontroller being instructed to send the hexadecimal number F3 to port A and 2C to port B:

Microcontroller

($F3)
Port A

($2C)
Port B

7 6 5 4 3 2 1 0   7 6 5 4 3 2 1 0
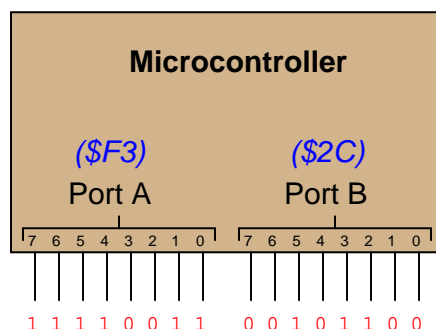
1 1 1 1 0 0 1 1   0 0 1 0 1 1 0 0

Suppose we wished to use the upper four bits of port A (pins 7, 6, 5, and 4) to drive the coils of a stepper motor in this eight-step sequence:

Step 1: 0001 10
Step 2: 0011 30
Step 3: 0010 20
Step 4: 0110 60
Step 5: 0100 40
Step 6: 1100 C0
Step 7: 1000 80
Step 8: 1001 90

As each pin goes high, it drives a power MOSFET on, which sends current through that respective coil of the stepper motor. By following a "shift" sequence as shown, the motor will rotate a small amount for each cycle.

Write the necessary sequence of numbers to be sent to port A to generate this specific order of bit shifts, in hexadecimal. Leave the lower four bit of port A all in the low logic state.

file 02895

Question 9

Convert the following numbers (all between the values of 0 and 1) into decimal form:

- $0.001_2 =$ .125
- $0.101_2 =$ .625
- $0.10111_2 =$ .719
- $0.005_8 =$
- $0.347_8 =$
- $0.34071_8 =$
- $0.00C_{16} =$
- $0.A2F_{16} =$
- $0.A2F09_{16} =$

file 01209

Question 10

Complete this table, performing all necessary conversions between numeration systems. Truncate all answers to three characters past the point:

Binary	Octal	Decimal	Hexadecimal
101.011			
		25.2	
			4.B
	72.52		
1011.101			
			AC.11
		934.79	
	641.7		
101100.1			

<u>file 01210</u>

Question 11

Counting practice: count from zero to thirty-one in binary, octal, and hexadecimal:

	Binary	Octal	Hex
Zero			
One			
Two			
Three			
Four			
Five			
Six			
Seven			
Eight			
Nine			
Ten			
Eleven			
Twelve			
Thirteen			
Fourteen			
Fifteen			

	Binary	Octal	Hex
Sixteen			
Seventeen			
Eighteen			
Nineteen			
Twenty			
Twenty one			
Twenty two			
Twenty three			
Twenty four			
Twenty five			
Twenty six			
Twenty seven			
Twenty eight			
Twenty nine			
Thirty			
Thirty one			

<u>file 01221</u>

7

Rotary encoders are electromechanical devices used to convert an angular position (shaft rotation) into a digital signal. The simplest form of rotary encoder uses a slotted wheel with a single LED/photodetector pair to generate pulses as the wheel turns:

Light sensor
(phototransistor)

LED
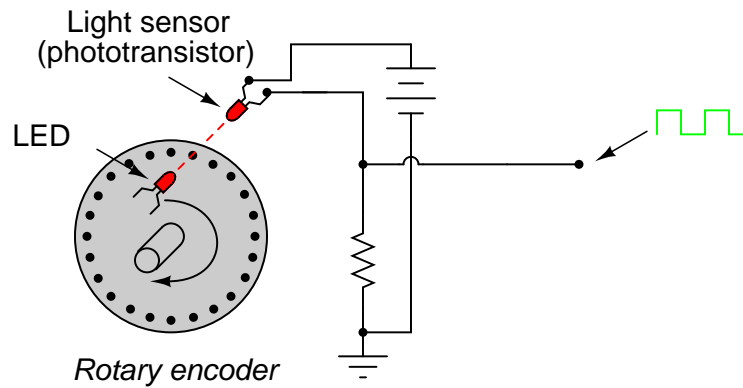
Rotary encoder

Some rotary encoder designs have multiple-bit outputs, with each LED/photodetector pair reading a different "track" of slots in the disk:
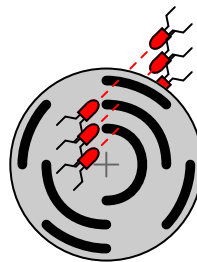
3-bit rotary encoder

In the illustration shown above, identify which LED/photodetector pairs represent the MSB (Most Significant Bit) and LSB (Least Significant Bit) of the binary output. Also, identify which direction the wheel must turn in order to produce an increasing count.

Note: assume that the darkest areas on the illustration represent slots cut through the disk, while the grey areas represent parts of the disk that are opaque.

file 01236

Question 13

Absolute rotary encoders often use a code known as *Gray code* rather than binary, to represent angular position. This code was patented by Frank Gray of Bell Labs in 1953, as a means of reducing errors in rotary encoder output. Examine each of these encoder disks, and determine which one is binary and which one is Gray code:
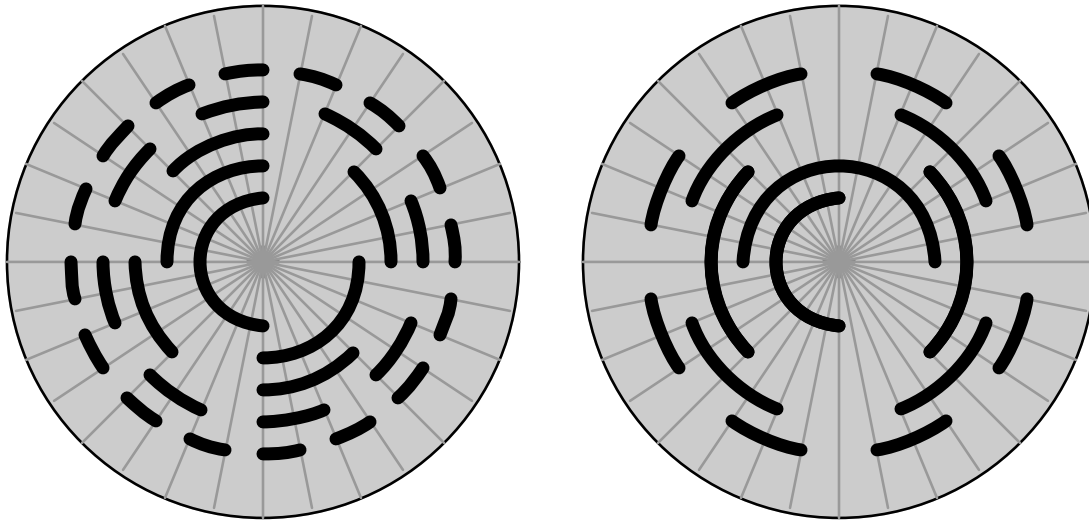
Assuming that the darkest areas on the illustration represent slots cut through the disk, and the grey areas represent parts of the disk that are opaque, mark the "zero," "one," and "two" sectors on each disk.

file 01237

Question 14

Explain why Gray code is often used in rotary encoders rather than binary coding. What difference does it make what type of code we use to mark the sectors of an encoder disk, so long as each sector possesses a unique number? Manufacturing errors can cause over/underlap. In binary this can cause MASSIVE problems

file 01238

Question 15

Convert the following binary numbers into Gray code: 0,1,3,2,6,7,5,4,C,D,F,E,A,B,9,8,0

- 100110_2 = 110101
- 110010_2 = 101011
- 101001_2 = 111101
- 1010100110_2 = 1111110101

file 01376

Question 16

Convert the following Gray code numbers into binary:

- 111110_{Gray} = 101011
- 100001_{Gray} = 111110
- 101110_{Gray} = 110100
- 1110001111_{Gray} = 1011110101

file 01377

Question 17

A nearly universal standard for representing text data in digital form is the *ASCII* code. What does the acronym "ASCII" stand for, and what is the format of this code?

file 01233

Question 18

Decode this set of ASCII characters, to reveal a secret message (all codes given in hexadecimal format):

49 20 4C 6F 76 65 20 45 6C 65 63 74 72 6F 6E 69 63 73 21

I Love Electronics!

file 01234

Question 19

Explain why Binary Coded Decimal (BCD) is sometimes referred to as the "8421" code. Why is this code used at all?

file 01239

Question 20

Convert the following decimal numbers into BCD form:

- 739_{10} 0111 0011 1001
- 25_{10} 0010 0101
- 92241_{10} 1001 0010 0010 0100 0001

Convert the following BCD numbers into decimal form:

- 1000 1001 89
- 0100 0111 0110 476
- 0011 1000 0101 0001 3851

file 01240

Question 21

Counting practice: count from zero to thirty-one in binary, octal, and hexadecimal:

	Binary	Octal	Hex
Zero			
One			
Two			
Three			
Four			
Five			
Six			
Seven			
Eight			
Nine			
Ten			
Eleven			
Twelve			
Thirteen			
Fourteen			
Fifteen			

	Binary	Octal	Hex
Sixteen			
Seventeen			
Eighteen			
Nineteen			
Twenty			
Twenty one			
Twenty two			
Twenty three			
Twenty four			
Twenty five			
Twenty six			
Twenty seven			
Twenty eight			
Twenty nine			
Thirty			
Thirty one			

file 01221

Question 22

Add the following binary numbers:

$$10010 + 1100 = 11110$$

$$1011101 + 1000000 = 10011101$$

$$10011 + 1111101 = 10010000$$

$$10011001 + 100111 = 11000000$$

$$11000011 + 101111 = 11110010$$

$$1001100 + 1100101 = 10110001$$

file 01220

Question 23

If the numbers sixteen and nine are added in binary form, will the answer be any different than if the same quantities are added in decimal form? Explain. No.

file 01229

Question 24

What is the *one's complement* of a binary number? If you had to describe this principle to someone who just learned what binary numbers are, what would you say?

Determine the one's complement for the following binary numbers:

- 10001010_2 01110101{2}
- 11010111_2 00101000{2}
- 11110011_2 00001100{2}
- 11111111_2 00000000{2}
- 11111_2 00000{2}
- 00000000_2 11111111{2}
- 00000_2 11111{2}

file 01222

Question 25

Determine the *two's complement* of the binary number 01100101_2. Explain how you did the conversion, step by step. 10011011

Next, determine the two's complement representation of the quantity *five* for a digital system where all numbers are represented by four bits, and also for a digital system where all numbers are represented by eight bits (one *byte*). Identify the difference that "word length" (the number of bits allocated to represent quantities in a particular digital system) makes in determining the two's complement of any number.

file 01224 1010
11111010

Question 26

In a computer system that represents all integer quantities using two's complement form, the most significant bit has a negative place-weight. For an eight-bit system, the place weights are as follows:

$$\overline{-2^7} \quad \overline{2^6} \quad \overline{2^5} \quad \overline{2^4} \quad \overline{2^3} \quad \overline{2^2} \quad \overline{2^1} \quad \overline{2^0}$$

Given this place-weighting, convert the following eight-bit two's complement binary numbers into decimal form:

- $01000101_2 = $ 69{10}
- $01110000_2 = $ 112{10}
- $11000001_2 = $ -63{10}
- $10010111_2 = $ -105{10}
- $01010101_2 = $ 75{10}
- $10101010_2 = $ -76{10}
- $01100101_2 = $ 101{10}

file 01225

Question 27

In an eight-bit digital system, where all numbers are represented in two's complement form, what is the largest (most positive) quantity that may be represented with those eight bits? What is the smallest (most negative) quantity that may be represented? Express your answers in both binary (two's complement) and decimal form. 127, -128

file 01226

Question 28

Two's complement notation really shows its value in binary addition, where positive and negative quantities may be handled with equal ease. Add the following byte-long (8 bit) two's complement numbers together, and then convert all binary quantities into decimal form to verify the accuracy of the addition:

```
  00110101 53        01110110 118        00111101
+ 00001100 12      + 00000010   2      + 11111011
  01000001 65        01111000 120        Overflow
```

```
  00001010   10       11111110           11111110
+ 10010101 -107      + 11011101         + 11111101
  10011111  -97        Overflow           Overflow
```

file 01227

Question 29

Add the following eight-bit two's complement numbers together, and then convert all binary quantities into decimal form to verify the accuracy of the addition:

```
  10110111            00111101  61        11111011
+ 01110110          + 00111011  59      + 11111011
  Overflow            01111000 120        Overflow
```

```
  10000001            01111011           01111111
+ 10010001          + 00111101         + 10000001
  Overflow            Rollover           Overflow
```

file 01230

Question 30

How is it possible to tell that *overflow* has occurred in the addition of binary numbers, without converting the binary sums to decimal form and having a human being verify the answers?

file 01231

9 bits

Question 31

Identify each of these logic gates by name, and complete their respective truth tables:
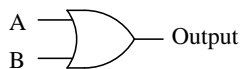
A —|>o— Output
B
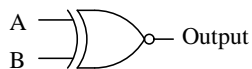
A	B	Output
0	0	
0	1	
1	0	
1	1	

A —)o— Output
B

A	B	Output
0	0	
0	1	
1	0	
1	1	

A —|)— Output
B

A	B	Output
0	0	
0	1	
1	0	
1	1	

A —))o— Output
B

A	B	Output
0	0	
0	1	
1	0	
1	1	

A —)— Output
B

A	B	Output
0	0	
0	1	
1	0	
1	1	

A —o|>— Output

A	Output
0	
1	

A —o|)— Output
B —o

A	B	Output
0	0	
0	1	
1	0	
1	1	

A —o)o— Output
B —o
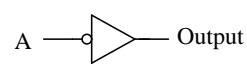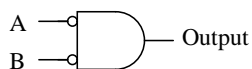
A	B	Output
0	0	
0	1	
1	0	
1	1	

A —)))— Output
B

A	B	Output
0	0	
0	1	
1	0	
1	1	

file 02776

Question 32

Add the following binary numbers:

```
    10010              1011101              10011
  + 1100            + 1000000          + 1111101
    11110             10011101           10010000
```

```
  10011001           11000011            1001100
+    100111        +    101111         + 1100101
  11000000           11110010           10110001
```

file 01220

14

Question 33

Design a circuit that adds two binary bits together. This circuit will have two inputs (A and B) and two outputs (Sum and Carry):

A ——[Adder]—— Sum

B ——[Adder]—— Carry

A B | Σ C
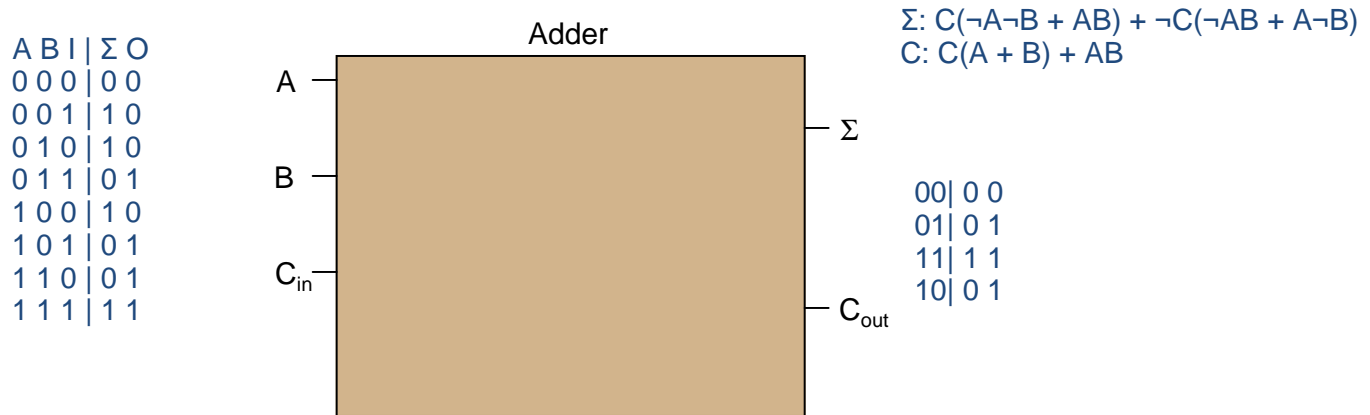0 0 | 0 0
0 1 | 1 0
1 0 | 1 0
1 1 | 0 1

Σ: XOR; C: AND
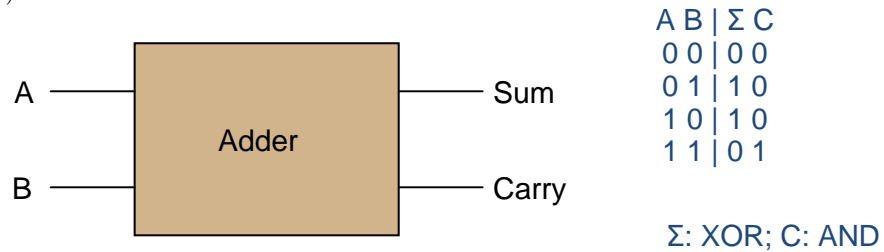
Begin the design process by drawing a truth table for the circuit, then determining the necessary gate circuitry to fulfill each output function.

Why is this circuit referred to as a *half adder*? How would a *full* adder circuit differ from this?
file 01478

Question 34

Design a circuit that adds two binary bits and a "Carry in" (C_{in}) bit together, producing a "Sum" (Σ) and a "Carry out" (C_{out}) output:

Σ: C(¬A¬B + AB) + ¬C(¬AB + A¬B)
C: C(A + B) + AB

A B I | Σ O
0 0 0 | 0 0
0 0 1 | 1 0
0 1 0 | 1 0
0 1 1 | 0 1
1 0 0 | 1 0
1 0 1 | 0 1
1 1 0 | 0 1
1 1 1 | 1 1

Adder

A ——

B ——

C_{in} ——

—— Σ

—— C_{out}

00| 0 0
01| 0 1
11| 1 1
10| 0 1

Begin the design process by drawing a truth table for the circuit, writing a boolean SOP expression for each output, then determining the necessary gate circuitry to fulfill each output function.
file 01479

Question 35

Explain the difference between a *ripple adder* and a *look-ahead* adder. What does the term "ripple" mean in this context? Why is "ripple" potentially a bad thing for a digital adder circuit?
file 02846

Ripple is slower. Propagation Delay.

15

Question 36

Compare the following two circuits, the first one being a digital adder and the second one being an analog summer:

A   B

0 1 2 3   0 1 2 3

74LS283

0 1 2 3

Σ

Digital Simpler
Analog faster & (usually) cheaper

R   R

V_A

R

TL071

R

V_Σ

V_B
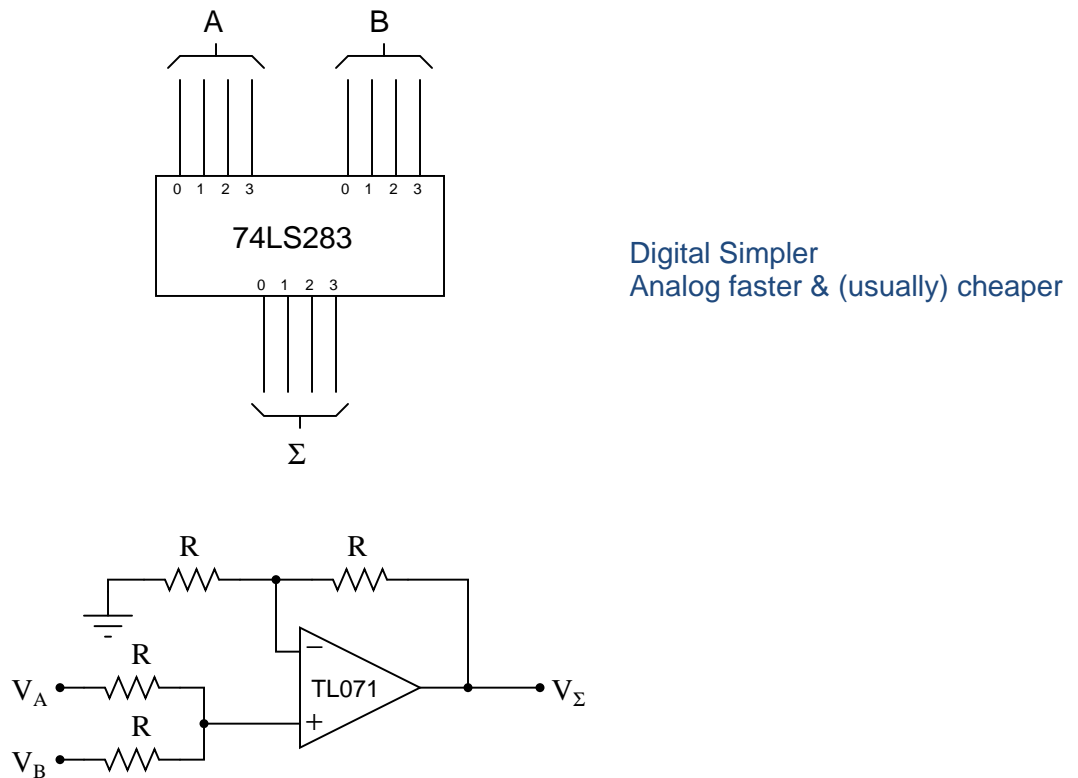
These two circuits perform the same mathematical function, yet the manners in which they perform this function are quite different. Compare and contrast the digital adder and the analog summer circuits shown here, citing any advantages or disadvantages of each.

file 02849

Question 37

Explain the purpose of a *magnitude comparator* IC such as the 74LS85. What function, or functions, does it perform?

file 02847

Question 38

Research the datasheet of an integrated arithmetic logic unit such as the 74AS181, and determine how its various modes of operation (addition, subtraction, comparison) are selected.

file 02845

An arithmetic trick often used when working with the metric system is multiplication-by-ten and division-by-ten via shifting of the decimal point. A similar "trick" may be applied to binary numbers, with similar results.

Determine what sort of multiplication or division is accomplished when the "binary point" is shifted in a binary number. Research the datasheet of an arithmetic logic unit (ALU) circuit to see if and how this function is implemented.

file 02850

Question 40

Explain the meaning of the digital lines A, B, F, and S in the following schematic diagram:
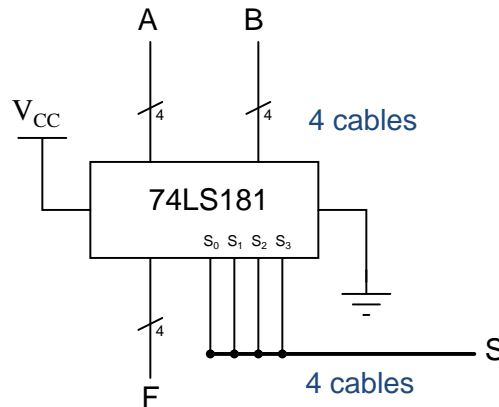
file 02848

Question 41

For a true TTL gate (not high-speed CMOS), what is the default logic state of an input line that is left floating (neither connected to V_{CC} nor Ground)? Explain why this is.   High

file 02863

Question 42

What logic state does a floating CMOS gate input naturally assume? How does this compare against traditional TTL?   Low

file 02865

Question 43

As an electronics instructor, I have the opportunity to see a lot of creative mistakes made by students as they learn to build circuits. One very common mistake made in CMOS circuit construction manifests itself in erratic behavior: the circuit may function correctly for a time, but suddenly and randomly it stops. Then, just by waving your hand next to the circuit, it begins to work again!

This problem is especially prevalent on days where the atmospheric humidity is low, and static electric charges easily accumulate on objects and people. Explain what sort of CMOS wiring mistake would cause a powered logic gate to behave erratically due to nearby static electric fields, and what the proper solution is to this problem.

file 01278   Use pulling resistors!

Question 44

Explain why the allowable power supply voltage range for a true TTL (not high-speed CMOS) logic gate is so narrow. What is the typical range of supply voltages for a true TTL gate, and why can't this type of logic gate operate from a wider range of voltages as CMOS gates can?

file 02864

Question 45

Logic probes are useful tools for troubleshooting digital logic gate circuits, but they certainly have limitations. For instance, in this simple circuit, a logic probe will give correct "high" and "low" readings at test point 1 (TP1), but it will always read "low" (even when the LED is on) at test point 2 (TP2):
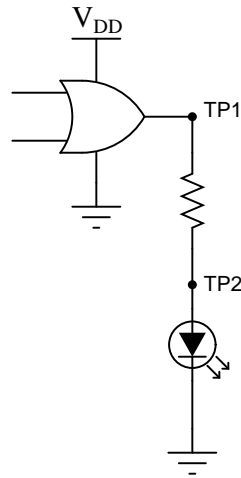
Now, obviously the output of the gate is "high" when the LED is on, otherwise it would not receive enough voltage to illuminate. Why then does a logic probe fail to indicate a high logic state at TP2?

file 02866

Question 46

A useful test instrument for digital gate circuit troubleshooting is a *logic pulser*. Explain what one is and give an example of how it is used.

file 02867

Question 47

A technician is using a logic pulser to force the logic state of the wire connecting two of the gates together:
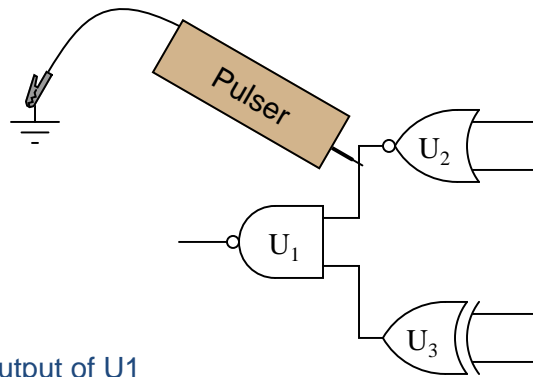
U1, probe on output of U1

Which gate, or gates, are we testing by placing the pulser in this position? What other instrument(s) would we have to connect to the circuit (and where?) to complete the test? Why does the logic pulser require a ground connection to do its job in this circuit?

file 02868

Question 48

The output of the following gate circuit is always low, no matter what states the input switches are in. Assume that CMOS logic gates are being used here:
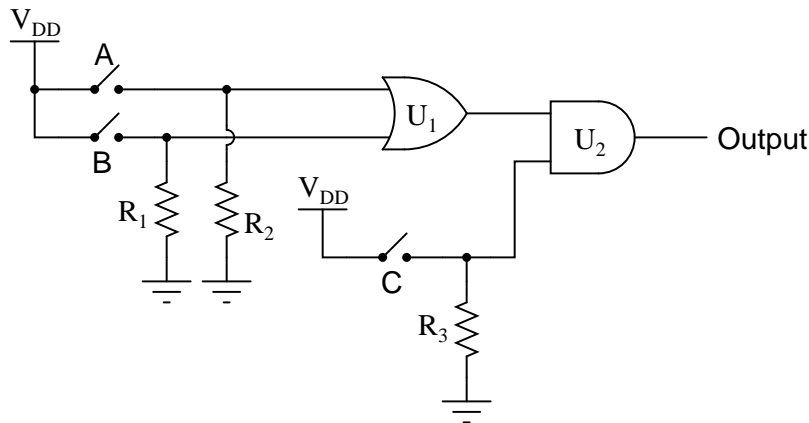
Identify which of these possibilities could account for the output always being low:

- Output of U_1 stuck in a high state NP
- Output of U_1 stuck in a low state P
- R_1 failed open NP
- Switch C failed open P
- Switch B failed open NP
- Switch A failed shorted NP

file 02892

19

Question 49

The output of the following gate circuit is always high, no matter what states the input switches are in. Assume that CMOS logic gates are being used here:
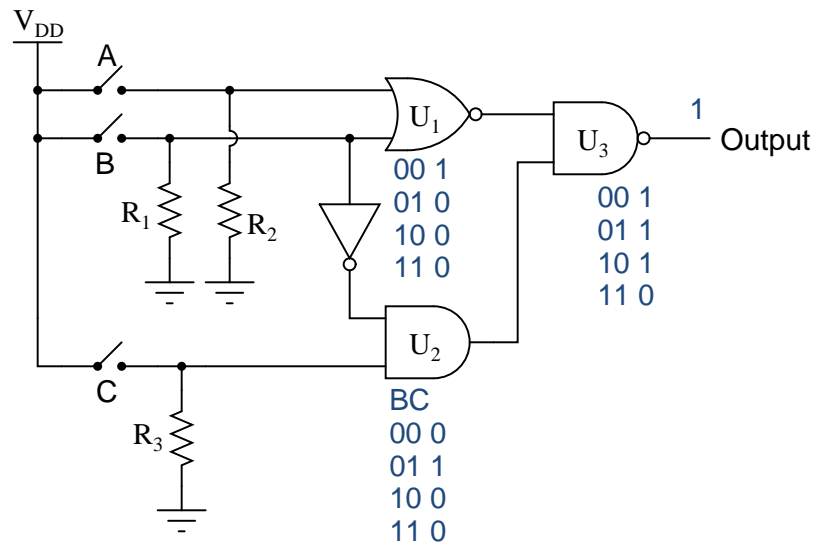
Identify which of these possibilities could account for the output always being high:

- Output of U_1 stuck in a high state NP
- Output of U_2 stuck in a high state NP
- R_1 failed open P?
- R_2 failed shorted NP
- R_3 failed shorted P
- Switch A failed open NP
- Switch B failed shorted P
- Switch C failed shorted NP

file 02894

20

The following gate circuit has a problem:

00 1
01 0
10 0
11 0

00 1
01 1
10 1
11 0
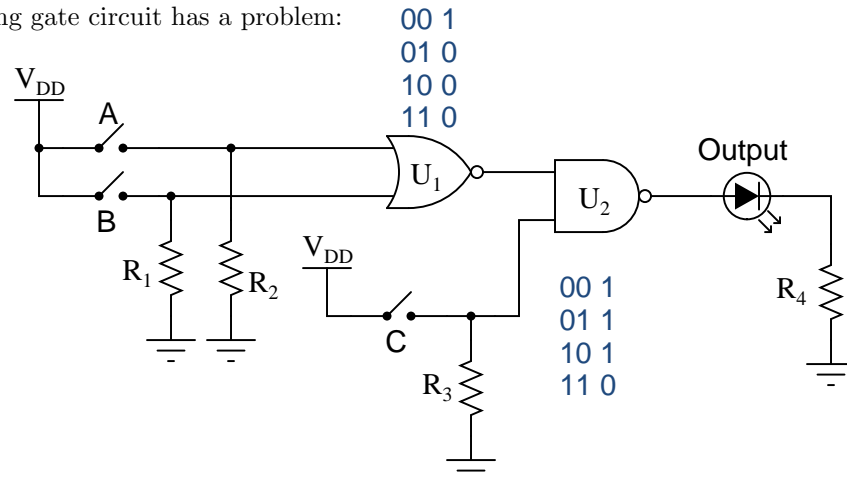
When tested, it is found that the circuit does not respond in the same manner as its (ideal) truth table predicts. Here is a comparison of the ideal and actual truth tables, as predicted and tested:

A	B	C	Output (ideal)	Output (actual)
0	0	0	1	1
0	0	1	0	0
0	1	0	1	1
0	1	1	1	0
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

The first thing a good electronics technician would do, of course, is set up either a voltmeter or a logic probe and begin testing for logic levels in the circuit to see what is wrong. However, the settings of the input switches are very important as part of the diagnosis. Based on the design of the circuit, and the truth table results shown, in what states (open or closed) would you first set the input switches, and then what logic level would you first test with the logic probe or voltmeter?

file 02949

A open, B & C closed
Test input B

The purpose of this circuit is to provide indication of when the rotary encoder shaft is in a particular position (matching the setting of the 8-position switch array):
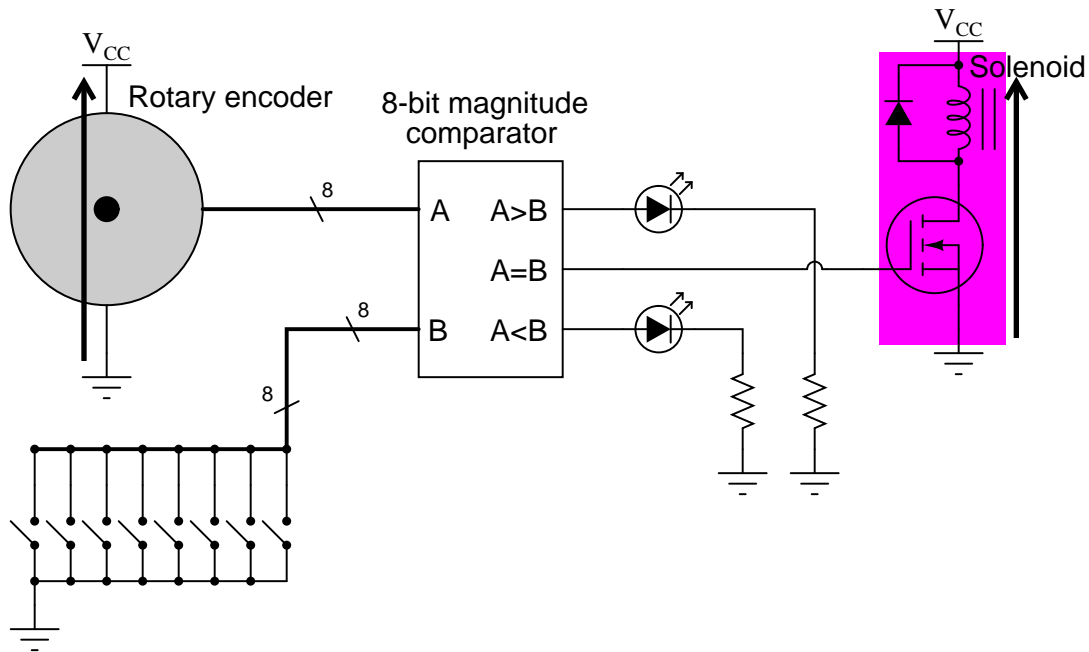
Trace the directions of all currents in this circuit, *using electron flow notation*, when the encoder position matches the pre-set code entered at the switches. Then, identify specific component failures that could result in the solenoid coil not energizing in this condition.

file 03845

Question 52

In this circuit, a comparator is set up to detect whether the outside or inside temperature is greater, and turns on a cooling fan when conditions are right. Predict how the operation of this circuit will be affected as a result of the following faults. Consider each fault independently (i.e. one at a time, no multiple faults):
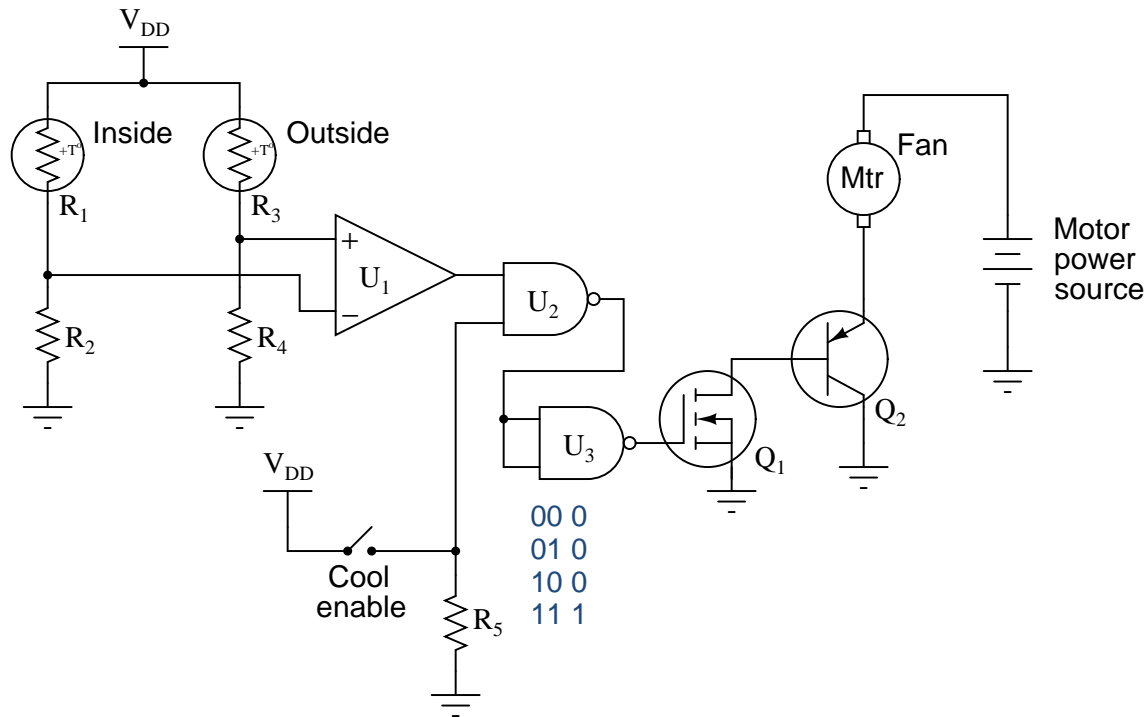

- Comparator U_1 output fails low: Fan always OFF
- NAND gate U_2 output fails low: Fan always ON
- NAND gate U_2 output fails high: Fan always OFF
- Transistor Q_1 fails shorted (drain to source): Fan always ON
- Resistor R_2 fails open: Fan always OFF
- Thermistor R_3 fails open: Fan always OFF
- Resistor R_4 fails open: Fan always ON when Cool switch CLOSED
- Solder bridge (short) across thermistor R_1: Fan always OFF

file 03846

Question 53

This *bargraph driver* circuit takes an audio input signal and displays the amplitude as a moving "bar" of lights. The stronger the amplitude of the signal, the more LEDs energize in the bargraph display. Predict how the operation of this circuit will be affected as a result of the following faults. Consider each fault independently (i.e. one at a time, no multiple faults):
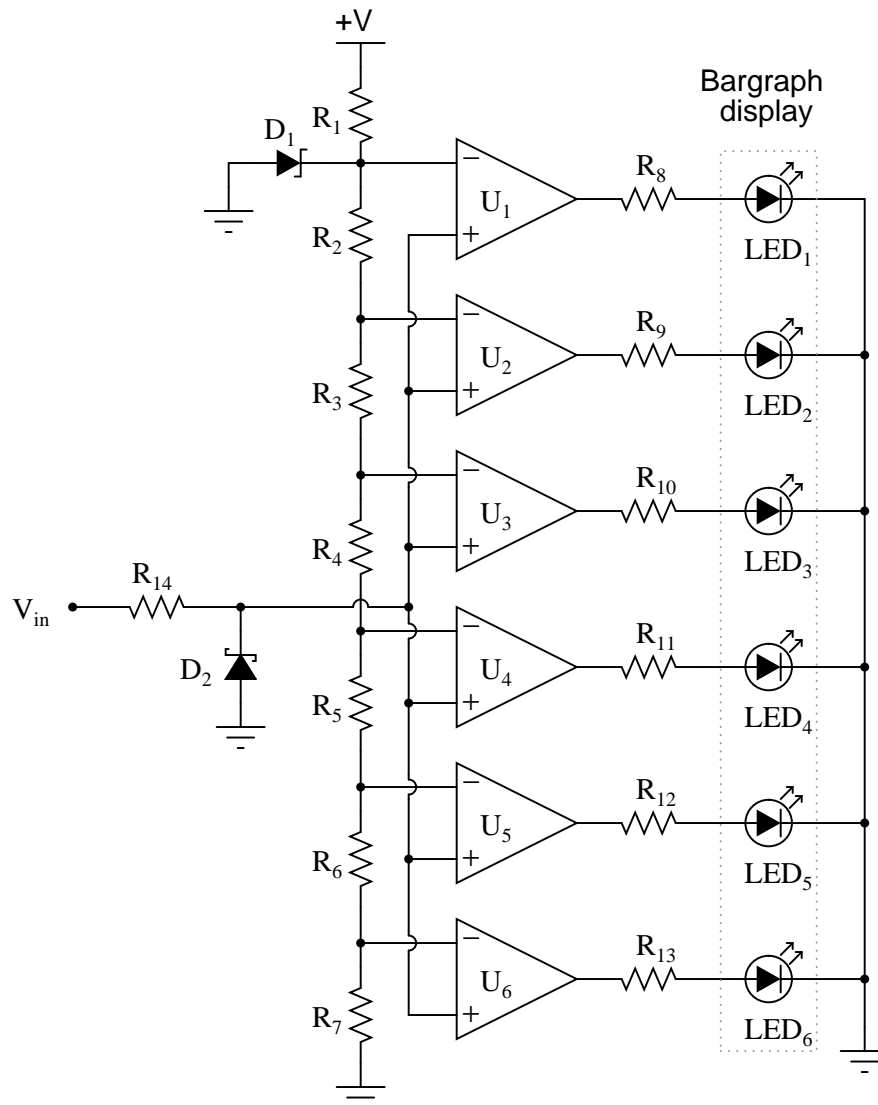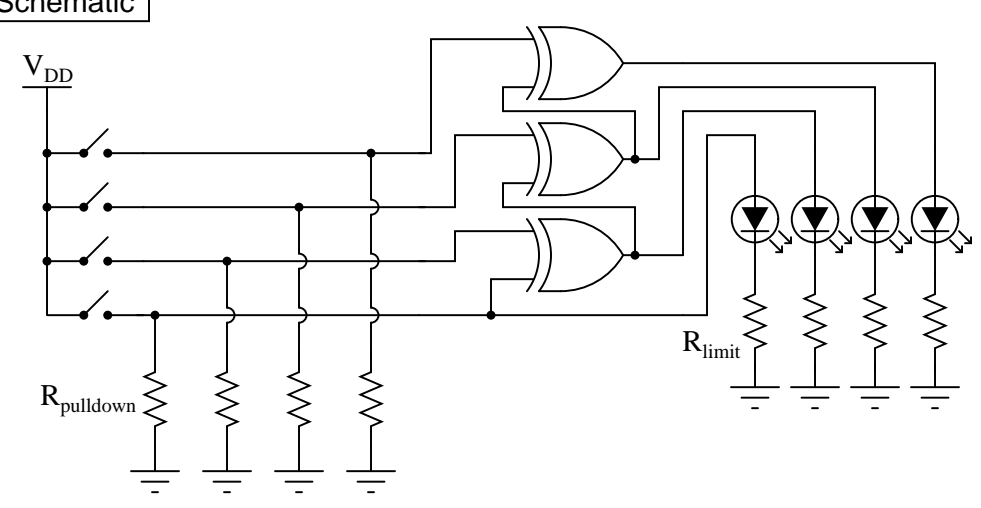

- Resistor R_4 failed open: LED 1 - 3 always OFF, LED 4 - 6 always ON (with audio in)
- Solder bridge (short) past resistor R_3: LED 2 & 3 always ON together
- Resistor R_{11} failed open: LED 4 always OFF
- Zener diode D_1 failed shorted: LED 1 - 6 always ON (with audio in)
- Schottky diode D_2 failed shorted: LED 1 - 6 always OFF

file 03848

Question 54

This *digital-to-analog* converter (DAC) circuit takes a four-bit binary input (input terminals A through D) and converts it to an analog voltage (V_{out}). Predict how the operation of this circuit will be affected as a result of the following faults. Consider each fault independently (i.e. one at a time, no multiple faults):

CMOS bilateral switches

(Arrow points in direction of electron flow)
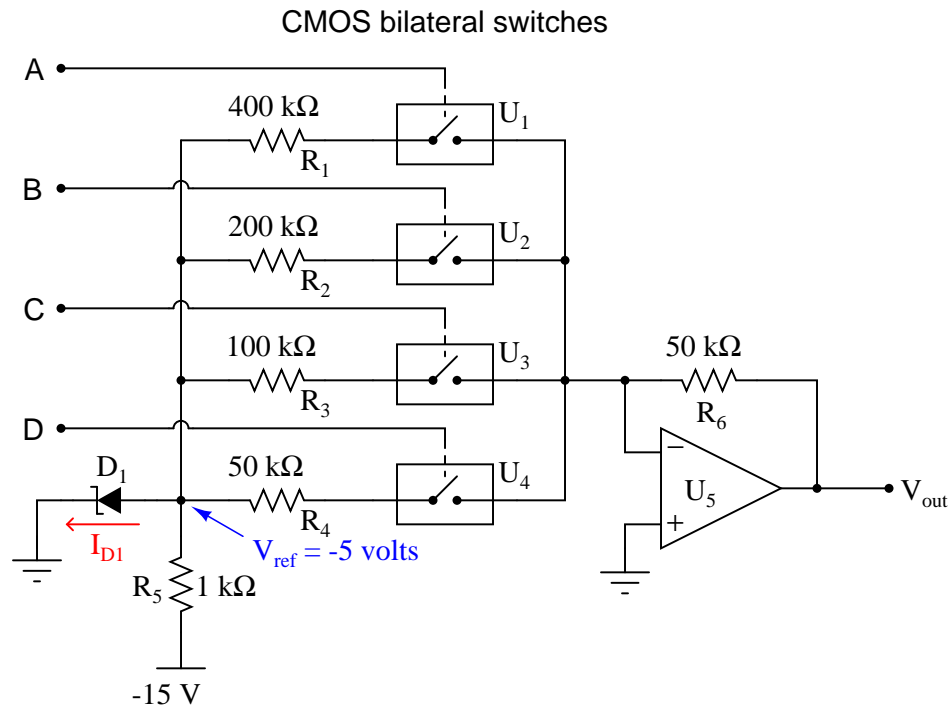
- Bilateral switch U_1 fails open: Changing input A does not change Vout
- Zener diode fails shorted: Vout is always 0V
- Solder bridge (short) past resistor R_1: Setting input A HIGH sends Vout to the upper rail
- Resistor R_6 fails open: U5 acts as a comparator

file 03849

25

Question 55

This is a digitally-set motor speed controller circuit, using PWM to modulate power to the motor. Predict how the operation of this circuit will be affected as a result of the following faults. Consider each fault independently (i.e. one at a time, no multiple faults):
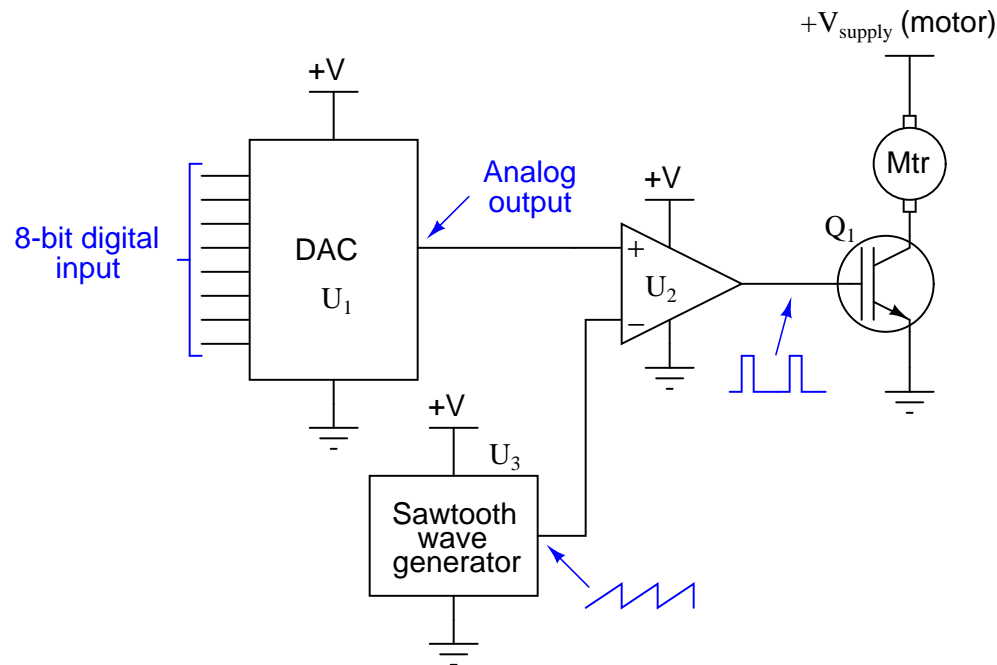
- DAC output fails low (output = 0 volts DC): Motor always OFF; No pulses from U2
- DAC output fails high (output = +V): Motor always ON
- IGBT Q_1 fails open (collector to emitter): Motor always OFF
- Solder bridge (short) between MSB input on U_1 and ground: Speeds 128 - 255 matched to speeds 1 -127

file 03850

Competency: **Gray to Binary code converter**   Version:

Schematic

Given conditions
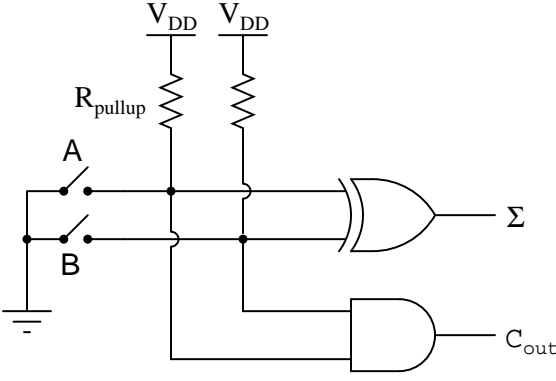
V_{DD} = 5V $R_{pulldown}$ = 10K R_{limit} = 180Ω

Truth table
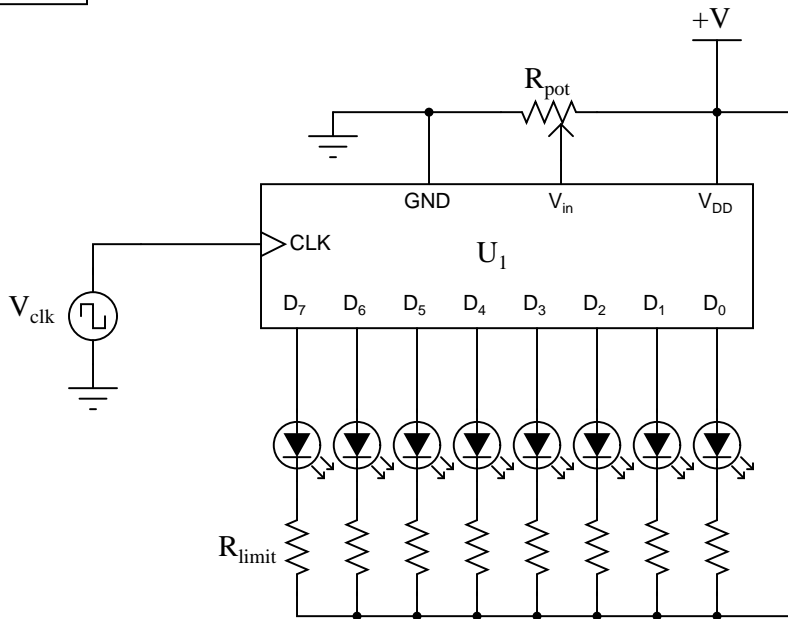
Predicted		Actual	
Gray	*Binary*	*Gray*	*Binary*
0 0 0 0	0000	0 0 0 0	0000
0 0 0 1	0001	0 0 0 1	0001
0 0 1 1	0010	0 0 1 1	0010
0 0 1 0	0011	0 0 1 0	0011
0 1 1 0	0100	0 1 1 0	0100
0 1 1 1	0101	0 1 1 1	0101
0 1 0 1	0110	0 1 0 1	0110
0 1 0 0	0111	0 1 0 0	0111
1 1 0 0	1000	1 1 0 0	1000
1 1 0 1	1001	1 1 0 1	1001
1 1 1 1	1010	1 1 1 1	1010
1 1 1 0	1011	1 1 1 0	1011
1 0 1 0	1100	1 0 1 0	1100
1 0 1 1	1101	1 0 1 1	1101
1 0 0 1	1110	1 0 0 1	1110
1 0 0 0	1111	1 0 0 0	1111

file 02856

Competency: **Half adder circuit**   Version:

Schematic

Given conditions

V_{DD} = 5V R_{pullup} = 10K R_{limit} = 330

Truth table

Predicted

A	B	C_{out}	Σ
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Actual

A	B	C_{out}	Σ
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

file 02857

Competency: **Full adder circuit w/ NAND gates**   Version:

Schematic

Given conditions

$V_{DD} = 5V$ $R_{pullup} = 10K$ $R_{limit} = 330$

Truth table

Predicted

A	B	C_{in}	C_{out}	Σ
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Actual

A	B	C_{in}	C_{out}	Σ
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

file 02859

Competency: **Analog-digital converter IC**   Version:

Description

Demonstrate the operation of an analog-to-digital
converter integrated circuit, using a potentiometer as
the variable input signal source.

Schematic

Parameters

	Predicted	Measured
$V_{resolution}$		

file 02950

Question 60

NAME: _____   **Project Grading Criteria**   PROJECT: _____

You will receive the highest score for which *all* criteria are met.

<u>100 %</u> *(Must meet or exceed all criteria listed)*
 A. Impeccable craftsmanship, comparable to that of a professional assembly
 B. No spelling or grammatical errors anywhere in *any* document, upon first submission to instructor

<u>95 %</u> *(Must meet or exceed these criteria in addition to all criteria for 90% and below)*
 A. Technical explanation sufficiently detailed to teach from, inclusive of every component (supersedes 75.B)
 B. Itemized parts list complete with part numbers, manufacturers, and (equivalent) prices for *all* components, including recycled components and parts kit components (supersedes 90.A)

<u>90 %</u> *(Must meet or exceed these criteria in addition to all criteria for 85% and below)*
 A. Itemized parts list complete with prices of components purchased for the project, plus total price
 B. No spelling or grammatical errors anywhere in *any* document upon final submission

<u>85 %</u> *(Must meet or exceed these criteria in addition to all criteria for 80% and below)*
 A. "User's guide" to project function (in addition to 75.B)
 B. Troubleshooting log describing all obstacles overcome during development and construction

<u>80 %</u> *(Must meet or exceed these criteria in addition to all criteria for 75% and below)*
 A. All controls (switches, knobs, etc.) clearly and neatly labeled
 B. All documentation created on computer, not hand-written (including the schematic diagram)

<u>75 %</u> *(Must meet or exceed these criteria in addition to all criteria for 70% and below)*
 A. Stranded wire used wherever wires are subject to vibration or bending
 B. Basic technical explanation of all major circuit sections
 C. Deadline met for working prototype of circuit (Date/Time = _____ / _____)

<u>70 %</u> *(Must meet or exceed these criteria in addition to all criteria for 65%)*
 A. All wire connections sound (solder joints, wire-wrap, terminal strips, and lugs are all connected properly)
 B. No use of glue where a fastener would be more appropriate
 C. Deadline met for submission of fully-functional project (Date/Time = _____ / _____) – supersedes 75.C if final project submitted by that (earlier) deadline

<u>65 %</u> *(Must meet or exceed these criteria in addition to all criteria for 60%)*
 A. Project fully functional
 B. All components securely fastened so nothing is "loose" inside the enclosure
 C. Schematic diagram of circuit

<u>60 %</u> *(Must meet or exceed these criteria in addition to being safe and legal)*
 A. Project minimally functional, with all components located inside an enclosure (if applicable)
 B. Passes final safety inspection (proper case grounding, line power fusing, power cords strain-relieved)

<u>0 %</u> *(If <u>any</u> of the following conditions are true)*
 A. Fails final safety inspection (improper grounding, fusing, and/or power cord strain relieving)
 B. Intended project function poses a safety hazard
 C. Project function violates any law, ordinance, or school policy
  <u>file 03173</u>

Question 61

Calculate the resistance between points **A** and **B** (R_{AB}) for the following resistor networks:

Figure 1
All resistors $500\ \Omega$
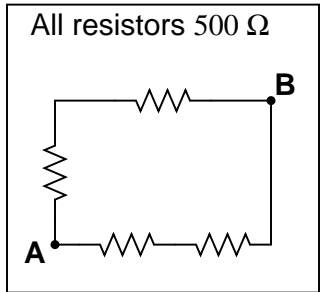
Figure 2
All resistors $1\ \text{k}\Omega$
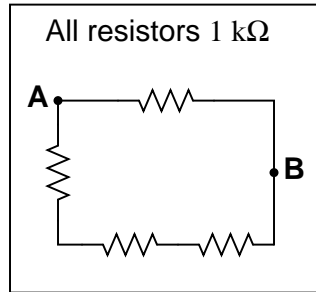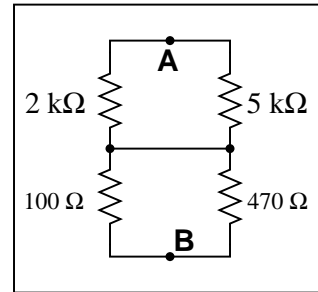
Figure 3
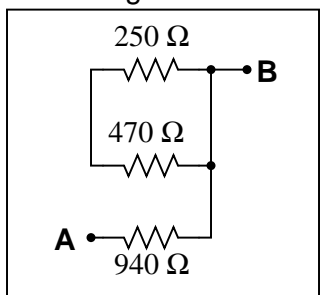
Figure 4

Figure 5
All resistors $2.2\ \text{k}\Omega$
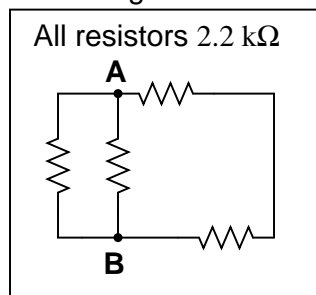
Figure 6
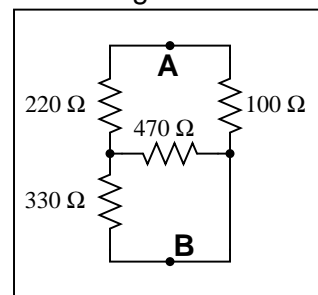
Question 62

Complete the table of values for this circuit:
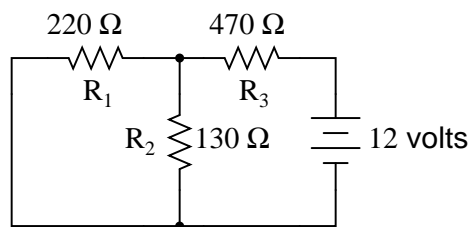
	R_1	R_2	R_3	Total
V				
I				
R	220 Ω	130 Ω	470 Ω	
P				

Question 63

Complete the table of values for this circuit:
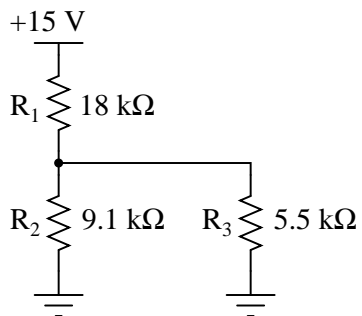
	R_1	R_2	R_3	Total
V				
I				
R	18 kΩ	9.1 kΩ	5.5 kΩ	
P				

Question 64

Complete the table of values for this circuit:
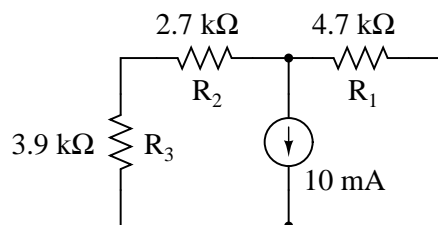
	R₁	R₂	R₃	Total
V				
I				10 mA
R	4.7 kΩ	2.7 kΩ	3.9 kΩ	
P				

Question 65

Complete the table of values for this circuit:
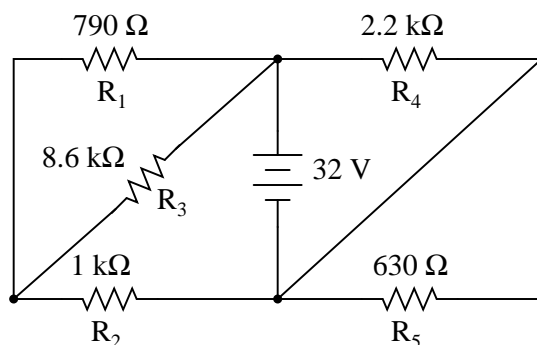
	R₁	R₂	R₃	R₄	R₅	Total
V						
I						
R	790 Ω	1 kΩ	8.6 kΩ	2.2 kΩ	630 Ω	
P						

Determine what a digital voltmeter (DVM) would indicate if connected between the following points in this circuit:
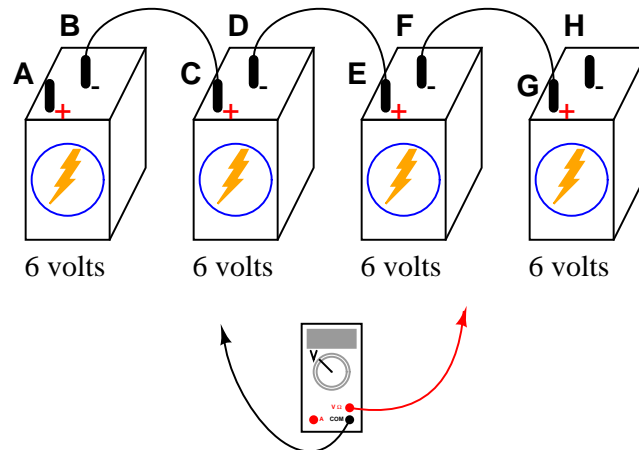
- Red lead on A, black lead on H
- Red lead on C, black lead on G
- Red lead on F, black lead on B
- Red lead on F, black lead on A

file 00344

Question 67

Identify each of the specified voltages in the following circuit. The subscripts refer to the specific test points (where the red test lead of the voltmeter touches the circuit), while ground is the point where the voltmeter's black lead is assumed to be attached:
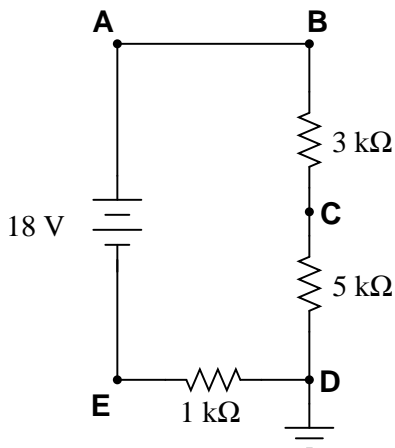
For example, V_B means the voltage indicated by a voltmeter with the red test lead touching point B and the black test lead touching ground.

- $V_A =$
- $V_B =$
- $V_C =$
- $V_D =$
- $V_E =$

file 03250

Question 68

Use Kirchhoff's Voltage Law to calculate the magnitude and polarity of the voltage across resistor R_4 in this resistor network:
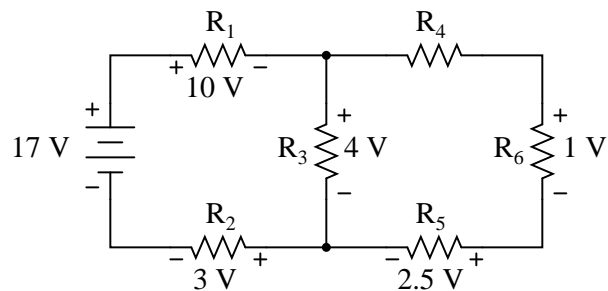
file 01728

Question 69

Calculate the amount of voltage between test points TP1 and TP3, and also the amount of voltage between test points TP2 and TP4:
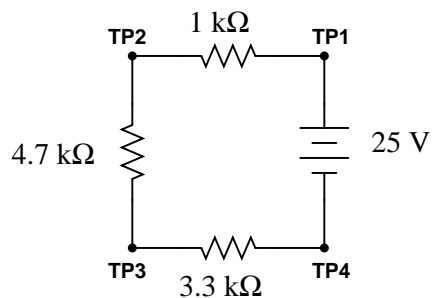
$V_{TP1-TP3} =$ $V_{TP2-TP4} =$

file 03251

Question 70

Determine the voltages registered by a voltmeter between the following points in this circuit:

$V_A =$ _____ (red lead on **A**, black lead on ground)

$V_B =$ _____ (red lead on **B**, black lead on ground)

$V_C =$ _____ (red lead on **C**, black lead on ground)

$V_D =$ _____ (red lead on **D**, black lead on ground)

$V_{AC} =$ _____ (red lead on **A**, black lead on **C**)

$V_{DB} =$ _____ (red lead on **D**, black lead on **B**)

$V_{BA} =$ _____ (red lead on **B**, black lead on **A**)

$V_{BC} =$ _____ (red lead on **B**, black lead on **C**)

$V_{CD} =$ _____ (red lead on **C**, black lead on **D**)

file 01961

37

Question 71

Determine the voltages registered by a voltmeter between the following points in this circuit:

C

4 V

3 V

B

13 V

A

6 V

D

$V_A = $ _____ (red lead on **A**, black lead on ground)

$V_B = $ _____ (red lead on **B**, black lead on ground)

$V_C = $ _____ (red lead on **C**, black lead on ground)

$V_D = $ _____ (red lead on **D**, black lead on ground)

$V_{AC} = $ _____ (red lead on **A**, black lead on **C**)

$V_{DB} = $ _____ (red lead on **D**, black lead on **B**)

$V_{BA} = $ _____ (red lead on **B**, black lead on **A**)

$V_{BC} = $ _____ (red lead on **B**, black lead on **C**)

$V_{CD} = $ _____ (red lead on **C**, black lead on **D**)

file 03844

Question 72

Draw a phasor diagram showing the trigonometric relationship between resistance, reactance, and impedance in this series circuit:

2.2 kΩ

R

5 V RMS
350 Hz

L

680 mH
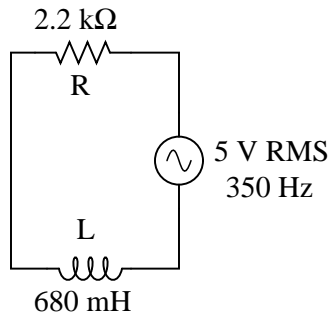
Show mathematically how the resistance and reactance combine in series to produce a total impedance (scalar quantities, all). Then, show how to analyze this same circuit using complex numbers: regarding component as having its own impedance, demonstrating mathematically how these impedances add up to comprise the total impedance (in both polar and rectangular forms).

file 01827

Question 73

Draw a phasor diagram showing the trigonometric relationship between resistance, reactance, and impedance in this series circuit:

2.2 kΩ

R

5 V RMS
350 Hz

C

0.22 μF
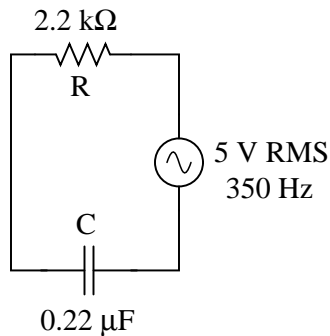
Show mathematically how the resistance and reactance combine in series to produce a total impedance (scalar quantities, all). Then, show how to analyze this same circuit using complex numbers: regarding each of the component as having its own impedance, demonstrating mathematically how these impedances add up to comprise the total impedance (in both polar and rectangular forms).

file 01828

Question 74

Calculate the total impedance of these parallel-connected components, expressing it in polar form (magnitude and phase angle):

Surface-mount components on a printed-circuit board

C1 33n   R1 510

Z_{total} @ 7.9 kHz = ???
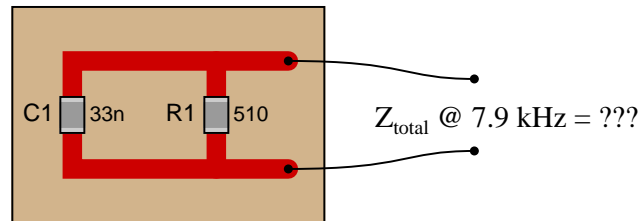
Also, draw an admittance triangle for this circuit.

file 02108

Question 75

Calculate the total impedance of this LR circuit, once using nothing but scalar numbers, and again using complex numbers:

R1
1k5

50m L1

Z_{total} @ 8 kHz = ???
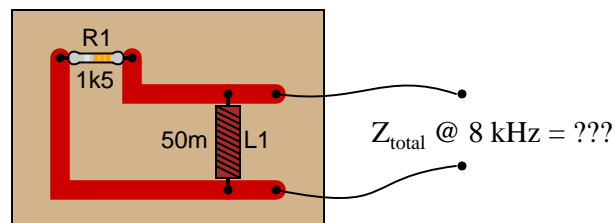
file 01837

Question 76

A student is asked to calculate the phase shift for the following circuit's output voltage, relative to the phase of the source voltage:
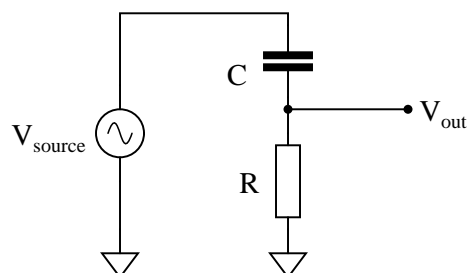
He recognizes this as a series circuit, and therefore realizes that a right triangle would be appropriate for representing component impedances and component voltage drops (because both impedance and voltage are quantities that add in series, and the triangle represents phasor addition):
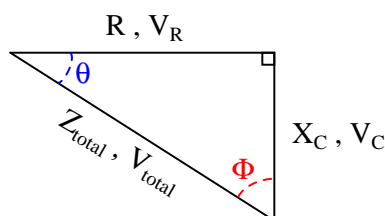
The problem now is, which angle does the student solve for in order to find the phase shift of V_{out}? The triangle contains two angles besides the 90° angle, Θ and Φ. Which one represents the output phase shift, and more importantly, *why*?

file 03748

Question 77

Calculate the output voltage of this phase-shifting circuit, expressing it in polar form (magnitude and phase angle relative to the source voltage):
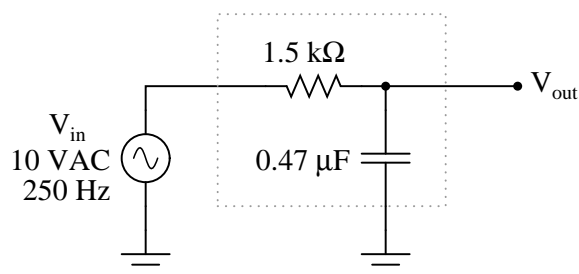
file 02620

Question 78

Determine the input frequency necessary to give the output voltage a phase shift of 70º:
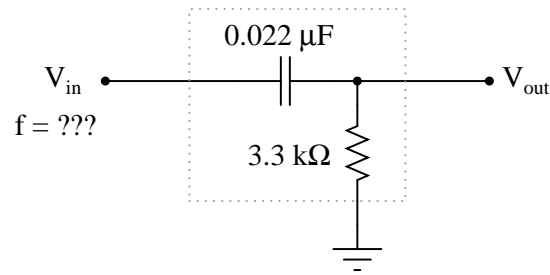
file 02623

Question 79

Determine the necessary resistor value to give the output voltage a phase shift of 44º:
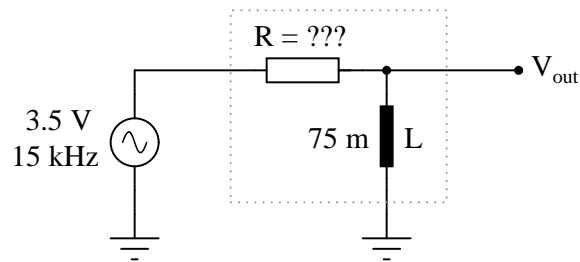
Also, write an equation that solves for this resistance value (R), given all the other variables $(f, L,$ and phase angle $\theta)$.

file 03283

Question 80

Determine the necessary resistor value to give the output voltage a phase shift of -64º:
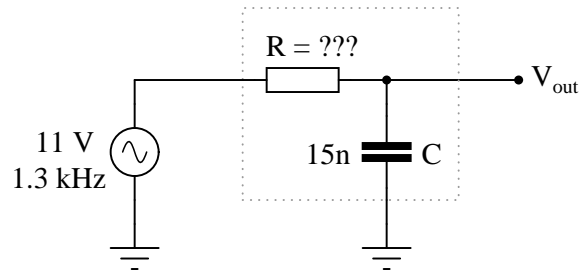
Also, write an equation that solves for this resistance value (R), given all the other variables $(f, C,$ and phase angle $\theta)$.

file 03287

The numeration system we use in our daily lives is called *base ten*, also called *decimal* or *denary*. What, exactly, does "base ten" mean?

Given the following base-ten number, identify which digits occupy the "one's place," "ten's place," "hundred's place," and "thousand's place," respectively:

$$5,183$$

<u>file 01195</u>

The ancient Mayans used a *vigesimal*, or base-twenty, numeration system in their mathematics. Each "digit" was a actually a composite of dots and/or lines, as such:
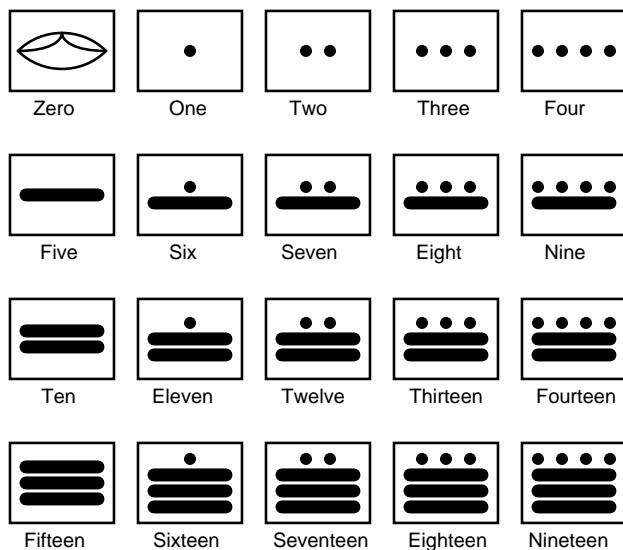
To represent numbers larger than twenty, the Mayans combined multiple "digits" the same way we do to represent numbers larger than ten. For example:
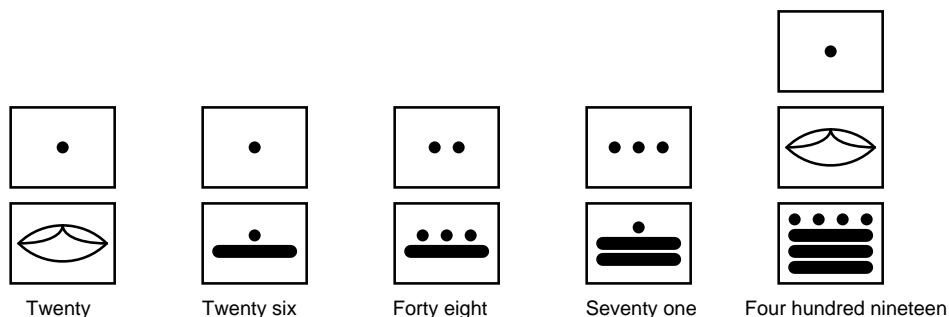
Based on the examples shown above, determine the place-weighting of each "digit" in the vigesimal numeration system. For example, in our denary, or base-ten, system, we have a one's place, a ten's place, a hundred's place, and so on, each successive "place" having ten times the "weight" of the place before it. What are the values of the respective "places" in the Mayan system?

Also, determine the values of these Mayan numbers:

44
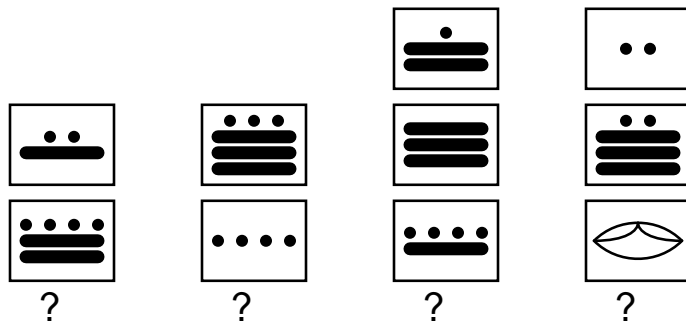
Question 83

If a digital meter display has four digits, it can represent any number from 0000 to 9999. This constitutes ten thousand unique numbers representable by the display. How many unique numbers could be represented by five digits? By six digits?

If an ancient Mayan ledger had spaces for writing numbers with three "digits" each, how many unique numbers could be represented in each space? What if the spaces were expanded to hold four "digits" each?

If a digital circuit has four *bits*, how many unique binary numbers can it represent? If we expanded its capabilities to eight bits, how many unique numbers could be represented by the circuit?

After answering these questions, do you see any mathematical pattern relating the number of "places" in a numeration system and the number of unique quantities that may be represented, given the "base" value ("radix") of the numeration system? Write a mathematical expression that solves for the number of unique quantities representable, given the "base" of the system and the number of "places".

file 01199

Question 84

How many binary bits are needed to count up to the number one million three hundred thousand seven hundred sixty two? Try to answer this question without converting this quantity into binary form, and then explain the mathematical procedure!

file 01228

The circuit shown in this diagram is used to transmit a numerical value from one location to another, by means of switches and lights:
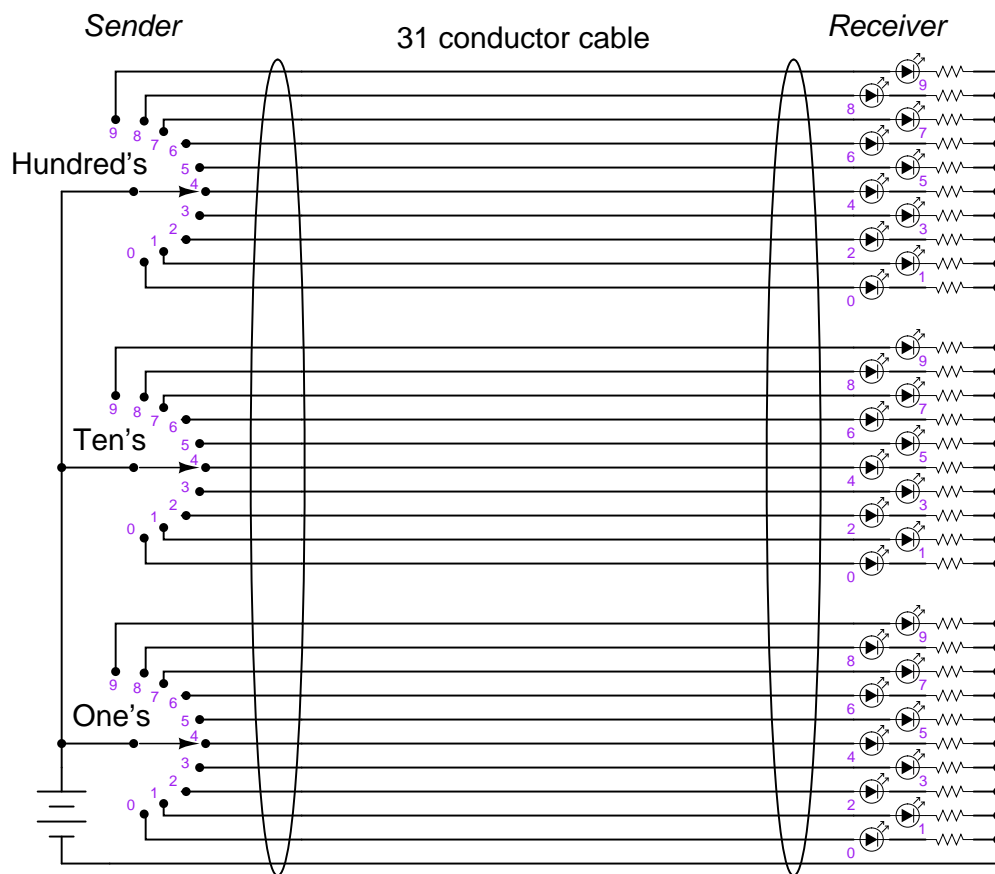
Given the switches and lights shown, any whole number between 0 and 999 may be transmitted from the switch location to the light location.

In fact, the arrangement shown here is not too different from an obsolete design of electronic base-ten indicators known as *Nixie tube* displays, where each digit was represented by a neon-filled glass tube in which one of ten distinct electrodes (each in the shape of a digit, 0-9) could be energized, providing glowing numerals for a person to view.

However, the system shown in the above diagram is somewhat wasteful of wiring. If we were to use the same thirty-one conductor cable, we could represent a much broader range of numbers if each conductor represented a distinct binary bit, and we used binary rather than base-ten for the numeration system:

Sender 31 conductor cable Receiver
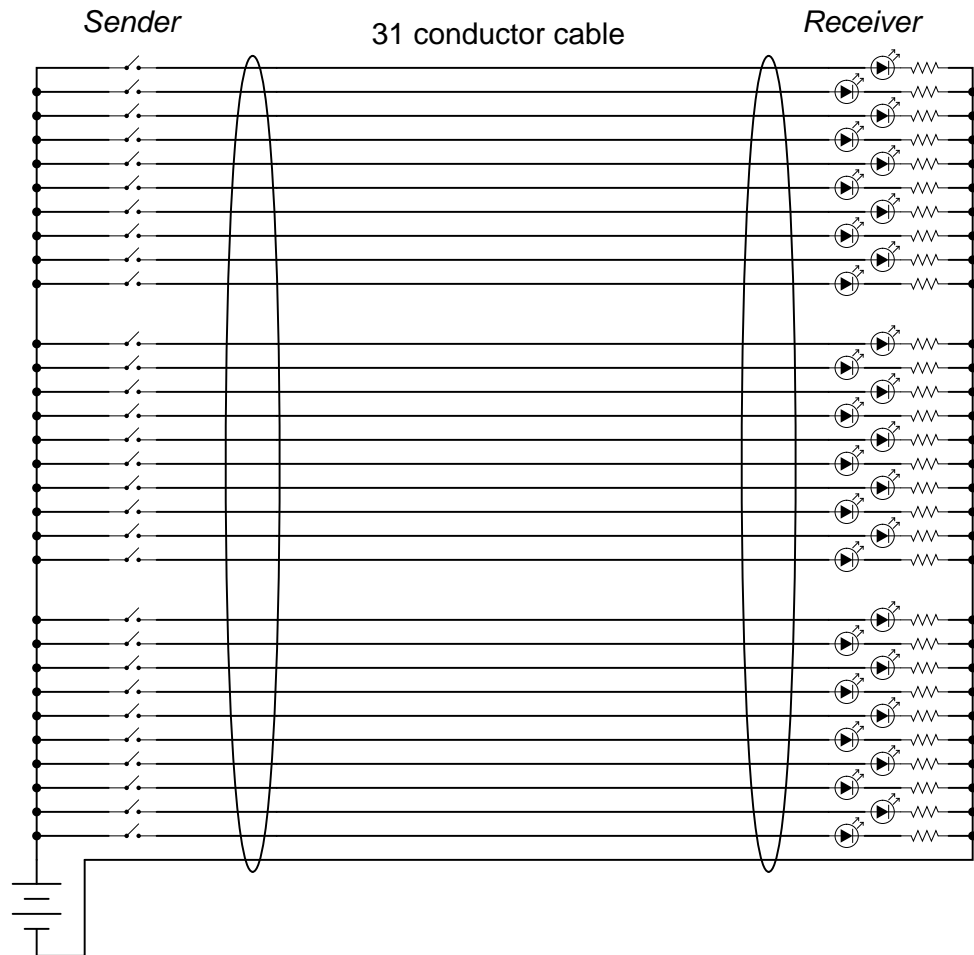
How many unique numbers are representable in this simple communications system? Also, what is the greatest individual number which may be sent from the "Sender" location to the "Receiver" location?

file 01201

Question 86

Explain why binary is a natural numeration system for expressing numbers in electronic circuits. Why not decimal or some other base of numeration?

How do you suspect binary numbers may be *stored* in electronic systems, for future retrieval? What advantages are there to the use of binary numeration in storage systems?

file 01203

When representing non-whole numbers, we extend the "places" of our decimal numeration system past the right of the decimal point, like this:

<div style="text-align:center">

Decimal place-weights

$$\frac{2}{10^3} \quad \frac{5}{10^2} \quad \frac{9}{10^1} \quad \frac{6}{10^0} \bullet \frac{3}{10^{-1}} \quad \frac{8}{10^{-2}} \quad \frac{0}{10^{-3}} \quad \frac{4}{10^{-4}}$$

</div>

$$2 \times 10^3 = 2000 \qquad\qquad 3 \times 10^{-1} = {}^3/_{10}$$
$$5 \times 10^2 = 500 \qquad\qquad 8 \times 10^{-2} = {}^8/_{100}$$
$$9 \times 10^1 = 90 \qquad\qquad 0 \times 10^{-3} = {}^0/_{1000}$$
$$6 \times 10^0 = 6 \qquad\qquad 4 \times 10^{-4} = {}^4/_{10000}$$

How do you suppose we represent non-whole numbers in a numeration system with a base (or "radix") other than ten? In the following examples, write the place-weight values underneath each place, and then determine the decimal equivalent of each example number:

<div style="text-align:center">

Binary place-weights

$$\underline{1} \quad \underline{0} \quad \underline{0} \quad \underline{1} \bullet \underline{1} \quad \underline{0} \quad \underline{1} \quad \underline{1}$$

</div>

<div style="text-align:center">

Octal place-weights

$$\underline{4} \quad \underline{0} \quad \underline{2} \quad \underline{7} \bullet \underline{3} \quad \underline{6} \quad \underline{1} \quad \underline{2}$$

</div>

<div style="text-align:center">

Hexadecimal place-weights

$$\underline{C} \quad \underline{1} \quad \underline{A} \quad \underline{6} \bullet \underline{3} \quad \underline{2} \quad \underline{B} \quad \underline{9}$$

</div>

Question 88

In digital computer systems, binary numbers are often represented by a fixed number of bits, such as 8, or 16, or 32. Such bit groupings are often given special names, because they are so common in digital systems:

- byte
- nybble
- word

How many binary bits is represented by each of the above terms?

And, for those looking for more challenge, try defining these terms:

- nickle
- deckle
- chawmp
- playte
- dynner

file 01223

Question 89

Explain how the *Morse Code* resembles ASCII in structure and purpose.
file 01241

Question 90

Standard ASCII codes are seven bits in length, but communications equipment usually sends data in eight-bit (byte) groups. The extra bit is often used as a *parity bit*. What is the purpose of a "parity bit," and how is it used?
file 01235

Question 91

What is a *floating-point* number in a digital system?
file 01232

Answer 1

There are only two valid ciphers in the binary system: 0 and 1. Each successive place carries twice the "weight" of the one before it. Seventeen = 10001 (in binary). Each character in the binary system is called a "bit" rather than a "digit".

Answer 2

999999_{10} = Nine hundred ninety nine thousand, nine hundred ninety nine.
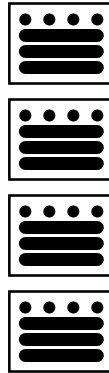
One hundred fifty nine thousand nine hundred ninety nine

1111111111_2 = one thousand twenty three.

Answer 3

- $10_2 = 2_{10}$
- $1010_2 = 10_{10}$
- $10011_2 = 19_{10}$
- $11100_2 = 28_{10}$
- $10111_2 = 23_{10}$
- $101011_2 = 43_{10}$
- $11100110_2 = 230_{10}$
- $10001101011_2 = 1131_{10}$

Answer 4

- $7_{10} = 111_2$
- $10_{10} = 1010_2$
- $19_{10} = 10011_2$
- $250_{10} = 11111010_2$
- $511_{10} = 111111111_2$
- $824_{10} = 1100111000_2$
- $1044_{10} = 10000010100_2$
- $9241_{10} = 10010000011001_2$

Answer 5

There are only eight valid ciphers in the octal system (0, 1, 2, 3, 4, 5, 6, and 7), with each successive place carrying eight times the "weight" of the place before it.

- 35_8 into decimal: 29_{10}
- 16_{10} into octal: 20_8
- 110010_2 into octal: 62_8
- 51_8 into binary: 101001_2

Follow-up question: why is octal considered a "shorthand" notation for binary numbers?

Answer 6

There are sixteen valid ciphers in the hexadecimal system (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F), with each successive place carrying sixteen times the "weight" of the place before it.

- 35_{16} into decimal: 53_{10}
- 34_{10} into hexadecimal: 22_{16}
- 11100010_2 into hexadecimal: $E2_{16}$
- 93_{16} into binary: 10010011_2

Follow-up question: why is hexadecimal considered a "shorthand" notation for binary numbers?

Answer 7

Binary	Octal	Decimal	Hexadecimal
10010	22	18	12
1011100	134	92	5C
11010	32	26	1A
110111	67	55	37
1100101	145	101	65
100100010	442	290	122
1111101000	1750	1000	3E8
11011110	336	222	DE
1011010110	1326	726	2D6

Step 1: 10_{16}
Step 2: 30_{16}
Step 3: 20_{16}
Step 4: 60_{16}
Step 5: 40_{16}
Step 6: $C0_{16}$
Step 7: 80_{16}
Step 8: 90_{16}

Follow-up question: write the same sequence in decimal rather than hexadecimal:

Step 1:
Step 2:
Step 3:
Step 4:
Step 5:
Step 6:
Step 7:
Step 8:

Answer 9

- $0.001_2 = 0.125_{10}$
- $0.101_2 = 0.625_{10}$
- $0.10111_2 = 0.71875_{10}$
- $0.005_8 = 0.009765625_{10}$
- $0.347_8 = 0.451171875_{10}$
- $0.34071_8 = 0.439239502_{10}$
- $0.00C_{16} = 0.002929688_{10}$
- $0.A2F_{16} = 0.636474609_{10}$
- $0.A2F09_{16} = 0.636483192_{10}$

Answer 10

Binary	Octal	Decimal	Hexadecimal
101.011	5.3	5.375	5.6
11001.001	31.146	25.2	19.333
100.101	4.54	4.687	4.B
111010.101	72.52	58.656	3A.A8
1011.101	13.5	11.625	B.A
10101100.000	254.042	172.066	AC.11
1110100110.110	1646.624	934.79	3A6.CA3
110100001.111	641.7	417.875	1A1.E
101100.1	54.4	44.5	2C.8

Answer 11

No answers given here – compare with your classmates!

Answer 12

I'll let you figure out the MSB, LSB, and up-count direction on your own! It isn't difficult to do if you have mastered counting in binary.
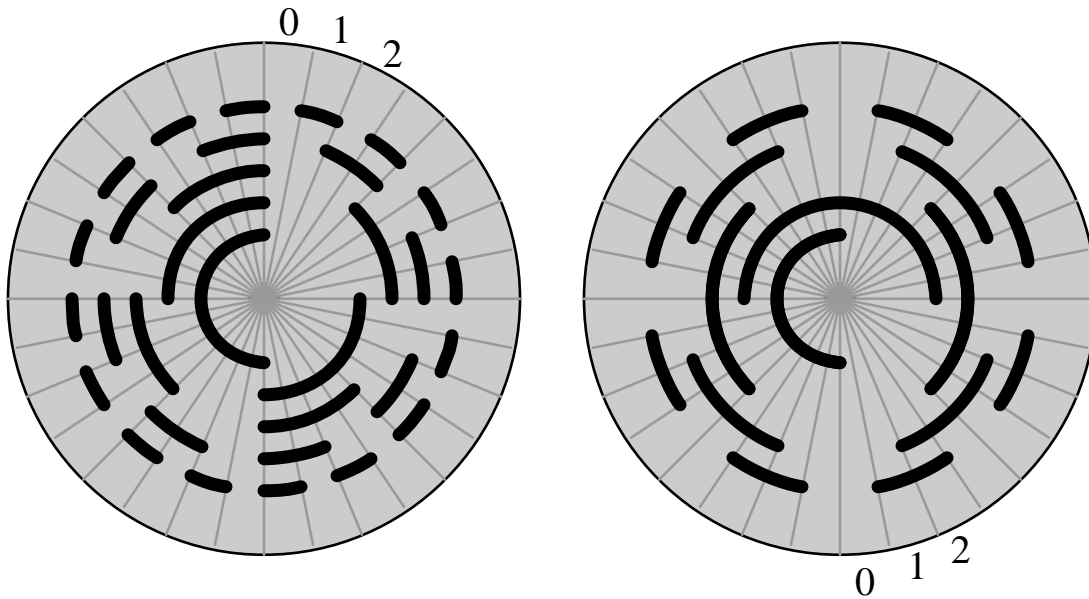
I won't directly tell you which disk is which, but I will provide a comparison of 5-bit binary versus Gray code, to help you in your analysis:

Binary	Gray
00000	00000
00001	00001
00010	00011
00011	00010
00100	00110
00101	00111
00110	00101
00111	00100
01000	01100
01001	01101
01010	01111
01011	01110
01100	01010
01101	01011
01110	01001
01111	01000
10000	11000
10001	11001
10010	11011
10011	11010
10100	11110
10101	11111
10110	11101
10111	11100
11000	10100

11001	10101
11010	10111
11011	10110
11100	10010
11101	10011
11110	10001
11111	10000

Answer 14

Gray code markings are more tolerant of sensor misalignment than binary markings, because there is no need for perfect synchronization of multiple bit transitions between sectors.

Answer 15

- $100110_2 = 110101_{\text{Gray}}$
- $110010_2 = 101011_{\text{Gray}}$
- $101001_2 = 111101_{\text{Gray}}$
- $1010100110_2 = 1111110101_{\text{Gray}}$

Answer 16

- $111110_{\text{Gray}} = 101011_2$
- $100001_{\text{Gray}} = 111110_2$
- $101110_{\text{Gray}} = 110100_2$
- $1110001111_{\text{Gray}} = 1011110101_2$

Answer 17

"ASCII" = American Standard Code for Information Interchange. Basic ASCII is a seven-bit binary code capable of representing all alphabetical characters used in the English language (upper-case as well as lower), as well as Arabic numerals, English punctuation marks, and some miscellaneous control codes for teletype machines.

Challenge question: although ASCII technically requires only 7 bits, a full 8 bits (1 byte) is usually reserved for each ASCII character in computer systems. Explain why.

Answer 18

I'll let you decode this message on your own!

Answer 19

BCD uses groups of four binary bits to represent each digit of a decimal number. The LSD place weights are 8-4-2-1, while the next significant digit's place weightings are 80-40-20-10, and so on.

Follow-up question: the four bits used for each BCD character could be called half of a byte (8 bits). There is a special word for a four-bit grouping. What is that word?

- $739_{10} = 0111\ 0011\ 1001$
- $25_{10} = 0010\ 0101$
- $92241_{10} = 1001\ 0010\ 0010\ 0100\ 0001$

- $1000\ 1001 = 89_{10}$
- $0100\ 0111\ 0110 = 476_{10}$
- $0011\ 1000\ 0101\ 0001 = 3851_{10}$

Answer 21

No answers given here – compare with your classmates!

Answer 22

```
    10010            1011101            10011
  + 1100          + 1000000          + 1111101
  ───────         ──────────         ──────────
    11110           10011101          10010000
```

```
  10011001          11000011          1001100
+   100111        +   101111        + 1100101
──────────        ──────────        ──────────
  11000000          11110010          10110001
```

Answer 23

No. The form of numeration used to represent numbers has no bearing on the outcome of mathematical operations.

Answer 24

- 10001010_2: One's complement $= 01110101_2$
- 11010111_2: One's complement $= 00101000_2$
- 11110011_2: One's complement $= 00001100_2$
- 11111111_2: One's complement $= 00000000_2$
- 11111_2: One's complement $= 00000_2$
- 00000000_2: One's complement $= 11111111_2$
- 00000_2: One's complement $= 11111_2$

Follow-up question: is the one's complement 11111111_2 identical to the one's complement of 11111_2? How about the one's complements of 00000000_2 and 00000_2? Explain.

Answer 25

The two's complement of 01100101 is 10011011.

The two's complement of five is 1011 in the four-bit system. It is 11111011 in the eight-bit system.

Answer 26

- $01000101_2 = 69_{10}$
- $01110000_2 = 112_{10}$
- $11000001_2 = -63_{10}$
- $10010111_2 = -105_{10}$
- $01010101_2 = 85_{10}$
- $10101010_2 = -86_{10}$
- $01100101_2 = 101_{10}$

Answer 27

Largest (most positive): $01111111_2 = 127_{10}$

Smallest (most negative): $10000000_2 = -128_{10}$

Answer 28

```
  00110101  53          01110110  118          00111101  61
+ 00001100  12        + 00000010    2        + 11111011  -5
  ────────              ────────               ────────
  01000001  65          01111000  120          00111000  56


  00001010  10          11111110  -2           11111110  -2
+ 10010101  -107      + 11011101  -35        + 11111101  -3
  ────────              ────────               ────────
  10011111  -97         11011011  -37          11111011  -5
```

Answer 29

```
  10110111  -73         00111101  61           11111011  -5
+ 01110110  118       + 00111011  59         + 11111011  -5
  ────────              ────────               ────────
  00101101  45          01111000  120          11110110  -10


  10000001  -127        01111011  123          01111111  127
+ 10010001  -111      + 00111101   61        + 10000001  -127
  ────────              ────────               ────────
  00010010  18          10111000  -72          00000000  0
```

Follow-up question: Why are some of these answers incorrect? Hint: perform the additions in *decimal* form rather than binary form, and then explain why those answers are not represented in the binary answers.

Answer 30

Check the sign bit of the answer, and compare it to the sign bits of the *addend* and *augend*.

Challenge question: under what condition(s) is overflow impossible? When can we add two binary numbers together and know with certainty that the answer will be correct?

Answer 31

NAND

A, B → Output

A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0

NOR
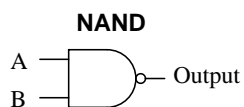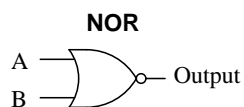
A, B → Output

A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0

AND
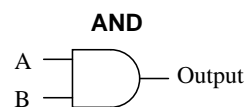
A, B → Output

A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1

XNOR

A, B → Output

A	B	Output
0	0	1
0	1	0
1	0	0
1	1	1

OR

A, B → Output

A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1

NOT

A → Output

A	Output
0	1
1	0

Neg-AND

A, B → Output

A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0

Neg-OR

A, B → Output

A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0

XOR

A, B → Output

A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

Answer 32

```
    10010           1011101            10011
  + 1100          + 1000000        + 1111101
  -------         ---------        ---------
    11110          10011101         10010000


  10011001         11000011          1001100
  +  100111        +  101111        + 1100101
  ---------        ---------        ---------
   11000000         11110010         10110001
```

58

Answer 33

Answer 34

$$\Sigma = \overline{A}\,\overline{B}C + \overline{A}\,B\overline{C} + A\overline{B}\,\overline{C} + ABC$$

$$C_{out} = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$
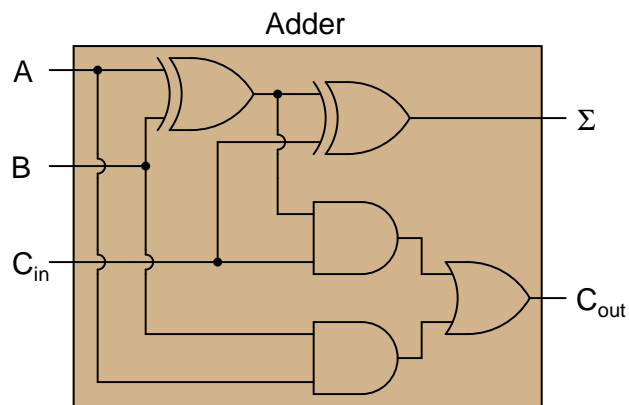
Answer 35

"Ripple" adders update their output bits one at a time rather than simultaneously. This leads to false, transient output states.

Answer 36

I won't directly give away answers here, but I will list a few criteria you might want to use for comparing and contrasting:

- Resolution
- Accuracy
- Speed
- Cost

Answer 37

I will let you research the datasheet for a magnitude comparator on your own to discover the answer(s)!

Answer 38

This is a small research project I leave up to you! Be sure to bring a copy of your IC datasheet to class for discussion!

Follow-up question: an interesting feature of the 74AS181 is that it provides "arithmetic" functions as well as "logic" functions. These two modes could also be referred to as "binary" and "boolean," respectively. Explain what distinguishes these two operating modes from one another, and why they are classified differently.

Answer 39

Shifting the "binary point" results in either multiplication or division by two. A multiplicative shift is performed by the 74AS181 ALU by arithmetic function selection 1100_2 (C_{16}).

Challenge question: explain how multiplication or division by *any* binary quantity may be accomplished using successive bit-shifts and additions. For example, show what steps you could take to multiply any binary number by five (101_2), using only "binary point" shifting and addition(s).

Answer 40

Lines A, B, and F (with the slash marks and the number "4") represent four actual conductors, carrying four bits of digital information. The thick line (S) is also a four-bit "bus" but is denoted by a slightly different convention.

In case you were wondering, it is unusual to mix two different bus symbol conventions in the same schematic diagram. I show this here only for your benefit, to see that there is more than one "standard" way to draw it.

Answer 41

Floating TTL inputs generally assume a "high" state due to the steering diode/resistor network on the input stage of each gate circuit.

Answer 42

A floating CMOS gate does not assume *any* definite logic state! The logic state of a floating CMOS gate input is indeterminate.

Follow-up question: what state does a floating input assume for a high-speed CMOS (74HCxx) logic gate, which is designed to be an upgrade/replacement for traditional TTL gates?

Answer 43

This classic problem is caused by a lack of pullup or pulldown resistors on CMOS gate inputs.

Answer 44

Due to the biasing requirements of its constituent bipolar transistors, TTL circuitry requires a much closer-regulated power supply voltage than CMOS. I'll let you research what this typical range is!

Answer 45

I won't give away the answer here, but it has something to do with proper CMOS logic level voltages.

Follow-up question: this LED circuit is rather simple, and the scenario almost silly, because the LED's presence makes checking the logic state at TP1 and TP2 superfluous! Can you think of any other circuit or situation where a similar false reading may be displayed by a logic probe – where the logic state has not been made visually obvious by the presence of an LED?

Answer 46

Since most digital electronics textbooks discuss logic pulsers along with logic probes, I'll let you do the research here and share with your classmates and instructor what you found!

Answer 47

In this location, the pulser is set up to test gate U_1. We would have to use a logic probe with "pulse" indication capability on the output of U_1 to complete the test.

The pulser requires a ground connection so it may drive current into or out of the circuit under test. Without a ground connection, there would be no complete path for current, and the pulser would not be able to "override" the output state of the NOR gate.

Follow-up question: what logic state should the *other* input of the NAND gate be in for this test? Explain why.

Answer 48

Only these two possibilities could account for the output always being low:

- Output of U_1 stuck in a low state
- Switch C failed open

Follow-up question: determine what each of the other faults would do to the circuit.

Answer 49

Only these three possibilities could account for the output always being high:

- R_1 failed open
- R_3 failed shorted
- Switch B failed shorted

Answer 50

Switch settings:

- A open (0)
- B closed (1)
- C closed (1)

Then, measure the logic state of the lower input on the NOR gate (coming from the "B" switch).
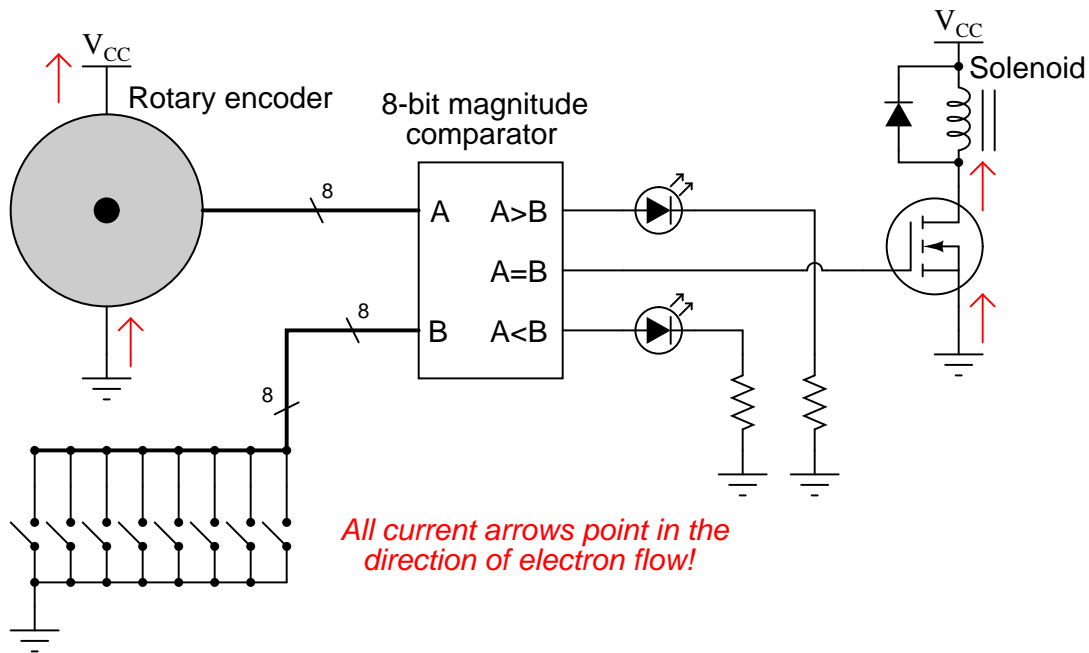
Potential component faults resulting in non-energization of solenoid:

- MOSFET failed open (drain-to-source)
- Broken wire between MOSFET drain and solenoid coil
- Loss of power (V_{CC}) to comparator circuit
- Loss of power to encoder (only if switch code is *not* 0000 0000)

Answer 52

- Comparator U_1 output fails low: *Fan never turns on.*
- NAND gate U_2 output fails low: *Fan always remains on.*
- NAND gate U_2 output fails high: *Fan never turns on.*
- Transistor Q_1 fails shorted (drain to source): *Fan always remains on.*
- Resistor R_2 fails open: *Fan never turns on.*
- Thermistor R_3 fails open: *Fan never turns on.*
- Resistor R_4 fails open: *Fan always remains on.*
- Solder bridge (short) across thermistor R_1: *Fan never turns on.*

Follow-up question: does Q_1 source or sink current? What about Q_2?

Answer 53

- Resistor R_4 failed open: *LEDs 1 through 3 always off, LEDs 4 through 6 always on (with any substantial input signal amplitude at all).*
- Solder bridge (short) past resistor R_3: *LEDs 2 and 3 always energize at the same time.*
- Resistor R_{11} failed open: *LED 4 never lights up.*
- Zener diode D_1 failed shorted: *All LEDs light up together with any substantial input signal amplitude at all.*
- Schottky diode D_2 failed shorted: *None of the LEDs ever light up.*

Follow-up question: does each comparator *source* or *sink* current to its respective LED?

Challenge question: if resistors R_1 through R_7 are all equal value, the response of the bargraph will be linear (twice the signal amplitude results in twice as many LEDs energized). What would have to be changed in this circuit to give the bargraph a *logarithmic* response, so it registered proportional to a decibel scale rather than a voltage scale?

Answer 54

- Bilateral switch U_1 fails open: V_{out} *same for all odd numbered input conditions as it is for next lowest even-numbered input condition (e.g. input value of 5 gives same output as input value of 4).*
- Zener diode fails shorted: V_{out} *is nearly zero volts for any input condition.*
- Solder bridge (short) past resistor R_1: V_{out} *saturates positive for any given odd-valued input condition.*
- Resistor R_6 fails open: V_{out} *always saturated.*

Follow-up question #1: is the arrow showing zener diode current drawn in the direction of electron flow or conventional flow?

Follow-up question #2: which input bit is the most significant (MSB) and which is the least significant (LSB)?

Answer 55

- DAC output fails low (output = 0 volts DC): *Motor will not run.*
- DAC output fails high (output = +V): *Motor runs full speed all the time.*
- IGBT Q_1 fails open (collector to emitter): *Motor will not run.*
- Solder bridge (short) between MSB input on U_1 and ground: *Speeds 0 through 127 work normally, but speeds 128 through 255 just duplicate speeds 0 through 127, respectively.*

Answer 56

Use circuit simulation software to verify your predicted and actual truth tables.

Answer 57

Use circuit simulation software to verify your predicted and actual truth tables.

Answer 58

Use circuit simulation software to verify your predicted and actual truth tables.

Answer 59

Use circuit simulation software to verify your predicted and actual truth tables.

Answer 60

Be sure you meet with your instructor if you have any questions about what is expected for your project!

Answer 61

Figure 1:

$R_{AB} = 500\ \Omega$

Figure 2:

$R_{AB} = 750\ \Omega$

Figure 3:

$R_{AB} = 1.511\ \text{k}\Omega$

Figure 4:

$R_{AB} = 940\ \Omega$

Figure 5:

$R_{AB} = 880\ \Omega$

Figure 6:

$R_{AB} = 80.54\ \Omega$

Answer 62

	R_1	R_2	R_3	Total
V	1.778 V	1.778 V	10.22 V	12 V
I	8.079 mA	13.67 mA	21.75 mA	21.75 mA
R	220 Ω	130 Ω	470 Ω	551.7 Ω
P	14.36 mW	24.30 mW	222.3 mW	261.0 mW

Answer 63

	R_1	R_2	R_3	Total
V	12.6 V	2.4 V	2.4 V	15 V
I	700 µA	263.7 µA	436.3 µA	700 µA
R	18 kΩ	9.1 kΩ	5.5 kΩ	21.43 kΩ
P	8.82 mW	632.8 µW	1.05 mW	10.5 mW

Follow-up question: how much voltage is present at the node (junction point) where R_1, R_2, and R_3 all connect together, measured with reference to ground?

Answer 64
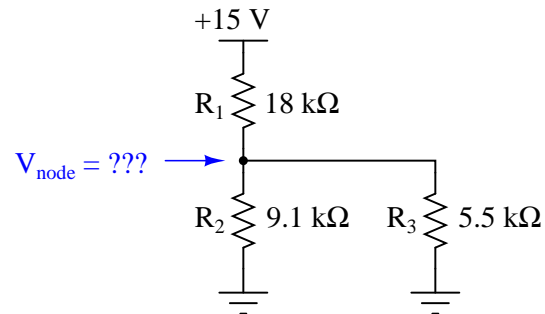
	R_1	R_2	R_3	Total
V	27.45 V	11.23 V	16.22 V	27.45 V
I	5.841 mA	4.159 mA	4.159 mA	10 mA
R	4.7 kΩ	2.7 kΩ	3.9 kΩ	2.745 kΩ
P	160.3 mW	46.71 mW	67.47 mW	274.5 mW

Answer 65

	R_1	R_2	R_3	R_4	R_5	Total
V	13.43 V	18.57 V	13.43 V	32 V	0 V	32 V
I	17.00 mA	18.57 mA	1.562 mA	14.55 mA	0 mA	33.11 mA
R	790 Ω	1 kΩ	8.6 kΩ	2.2 kΩ	630 Ω	966.4 Ω
P	228.4 mW	344.7 mW	20.98 mW	465.5 mW	0 mW	1.06 W

Challenge question: what circuit parameters will change if the diagonal wire in the right-hand side of the circuit is cut?

Answer 66
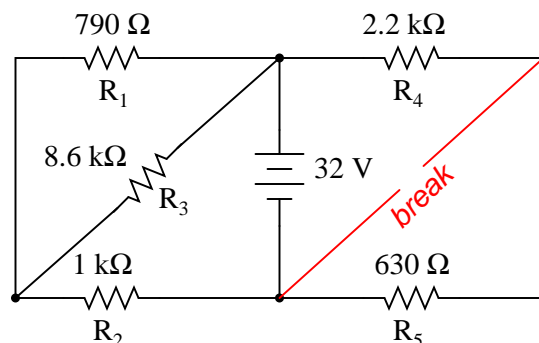
Here is a schematic diagram to help you:
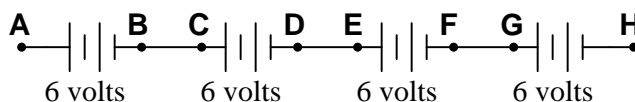
- Red lead on A, black lead on H = +24 volts
- Red lead on C, black lead on G = +12 volts
- Red lead on F, black lead on B = -12 volts
- Red lead on F, black lead on A = -18 volts

Answer 67

- V_A = +16 volts
- V_B = +16 volts
- V_C = +10 volts
- V_D = 0 volts
- V_E = -2 volts

Follow-up question: explain how it is possible to determine that V_A and V_B will be exactly the same value, prior to performing any mathematical calculations.

Answer 68

Answer 69
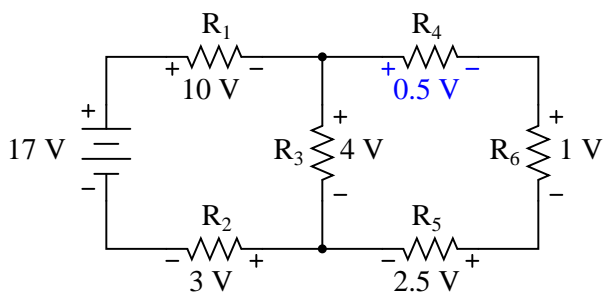
$V_{TP1-TP3} = 15.83$ volts $V_{TP2-TP4} = 22.22$ volts

Answer 70

$V_A = \underline{+12\text{ volts}}$ (red lead on **A**, black lead on ground)

$V_B = \underline{-9\text{ volts}}$ (red lead on **B**, black lead on ground)

$V_C = \underline{+4.5\text{ volts}}$ (red lead on **C**, black lead on ground)

$V_D = \underline{-24\text{ volts}}$ (red lead on **D**, black lead on ground)

$V_{AC} = \underline{+7.5\text{ volts}}$ (red lead on **A**, black lead on **C**)

$V_{DB} = \underline{-15\text{ volts}}$ (red lead on **D**, black lead on **B**)

$V_{BA} = \underline{-21\text{ volts}}$ (red lead on **B**, black lead on **A**)

$V_{BC} = \underline{-13.5\text{ volts}}$ (red lead on **B**, black lead on **C**)

$V_{CD} = \underline{+28.5\text{ volts}}$ (red lead on **C**, black lead on **D**)

Answer 71

$V_A = \underline{+13\text{ volts}}$ (red lead on **A**, black lead on ground)

$V_B = \underline{+4\text{ volts}}$ (red lead on **B**, black lead on ground)

$V_C = \underline{+7\text{ volts}}$ (red lead on **C**, black lead on ground)

$V_D = \underline{-6\text{ volts}}$ (red lead on **D**, black lead on ground)

$V_{AC} = \underline{+6\text{ volts}}$ (red lead on **A**, black lead on **C**)

$V_{DB} = \underline{-10\text{ volts}}$ (red lead on **D**, black lead on **B**)

$V_{BA} = \underline{-9\text{ volts}}$ (red lead on **B**, black lead on **A**)

$V_{BC} = \underline{-3\text{ volts}}$ (red lead on **B**, black lead on **C**)

$V_{CD} = \underline{+13\text{ volt}}$ (red lead on **C**, black lead on **D**)

Answer 72
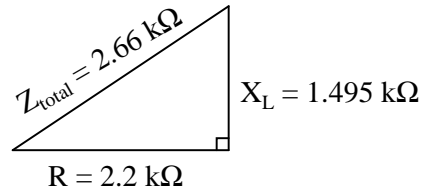
Scalar calculations

$R = 2.2 \text{ k}\Omega \qquad X_L = 1.495 \text{ k}\Omega$

$Z_{series} = \sqrt{R^2 + X_L{}^2}$

$Z_{series} = \sqrt{2200^2 + 1495^2} = 2660 \ \Omega$

Complex number calculations

$\mathbf{Z_R} = 2.2 \text{ k}\Omega \angle 0^o \qquad \mathbf{Z_L} = 1.495 \text{ k}\Omega \angle 90^o \quad$ (Polar form)

$\mathbf{Z_R} = 2.2 \text{ k}\Omega + j0 \ \Omega \qquad \mathbf{Z_L} = 0 \ \Omega + j1.495 \text{ k}\Omega \quad$ (Rectangular form)

$\mathbf{Z_{series}} = \mathbf{Z_1} + \mathbf{Z_2} + \cdots \mathbf{Z_n} \quad$ (General rule of series impedances)

$\mathbf{Z_{series}} = \mathbf{Z_R} + \mathbf{Z_L} \quad$ (Specific application to this circuit)

$\mathbf{Z_{series}} = 2.2 \text{ k}\Omega \angle 0^o + 1.495 \text{ k}\Omega \angle 90^o = 2.66 \text{ k}\Omega \angle 34.2^o$

$\mathbf{Z_{series}} = (2.2 \text{ k}\Omega + j0 \ \Omega) + (0 \ \Omega + j1.495 \text{ k}\Omega) = 2.2 \text{ k}\Omega + j1.495 \text{ k}\Omega$

Answer 73
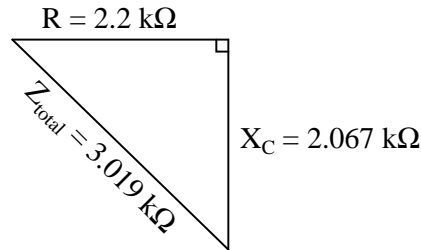
Scalar calculations

$R = 2.2 \text{ k}\Omega \qquad X_C = 2.067 \text{ k}\Omega$

$Z_{series} = \sqrt{R^2 + X_C{}^2}$

$Z_{series} = \sqrt{2200^2 + 2067^2} = 3019 \ \Omega$

Complex number calculations

$\mathbf{Z_R} = 2.2 \text{ k}\Omega \angle 0^o \qquad \mathbf{Z_C} = 2.067 \text{ k}\Omega \angle -90^o \quad$ (Polar form)

$\mathbf{Z_R} = 2.2 \text{ k}\Omega + j0 \ \Omega \qquad \mathbf{Z_C} = 0 \ \Omega - j2.067 \text{ k}\Omega \quad$ (Rectangular form)

$\mathbf{Z_{series}} = \mathbf{Z_1} + \mathbf{Z_2} + \cdots \mathbf{Z_n} \quad$ (General rule of series impedances)

$\mathbf{Z_{series}} = \mathbf{Z_R} + \mathbf{Z_C} \quad$ (Specific application to this circuit)

$\mathbf{Z_{series}} = 2.2 \text{ k}\Omega \angle 0^o + 2.067 \text{ k}\Omega \angle -90^o = 3.019 \text{ k}\Omega \angle -43.2^o$

$\mathbf{Z_{series}} = (2.2 \text{ k}\Omega + j0 \ \Omega) + (0 \ \Omega - j2.067 \text{ k}\Omega) = 2.2 \text{ k}\Omega - j2.067 \text{ k}\Omega$

Answer 74

$Z_{total} = 391.4 \ \Omega \ \angle \ \text{-}39.9^o$

$Y_{total} = 2.555$ mS

$Y_{C1} = 1.638$ mS

39.9^o

$Y_{R1} = 1.961$ mS
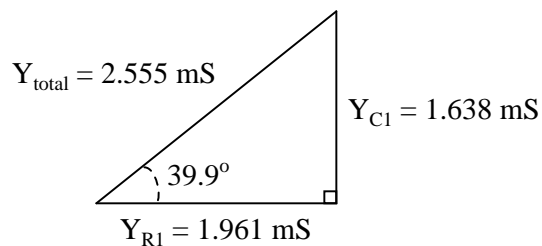
Answer 75

Scalar calculations

$R_1 = 1.5$ kΩ $G_{R1} = 666.7 \ \mu$S

$X_{L1} = 2.513$ kΩ $B_{L1} = 397.9 \ \mu$S

$Y_{total} = \sqrt{G^2 + B^2} = 776.4 \ \mu$S

$Z_{total} = \frac{1}{Y_{total}} = 1.288$ kΩ

Complex number calculations

$R_1 = 1.5$ kΩ $\mathbf{Z_{R1}} = 1.5$ kΩ $\angle \ 0^o$

$X_{L1} = 2.513$ kΩ $\mathbf{Z_{L1}} = 2.513$ kΩ $\angle \ 90^o$

$\mathbf{Z_{total}} = \frac{1}{\frac{1}{\mathbf{Z_{R1}}} + \frac{1}{\mathbf{Z_{L1}}}} = 1.288$ kΩ $\angle \ 30.83^o$

Answer 76

The proper angle in this circuit is Θ, and it will be a positive (leading) quantity.

Answer 77

$V_{out} = 6.7$ V $\angle \ \text{-}47.9^o$

Answer 78

$f = 798$ Hz

Answer 79

$R = 6.826$ kΩ

$$R = 2\pi f L \tan \theta$$

Answer 80

$R = 16.734$ kΩ

$$R = -\frac{\tan \theta}{2\pi f C}$$

Answer 81

"Base ten" means that numbers are represented by combinations of symbols (ciphers), of which there are only ten (0, 1, 2, 3, 4, 5, 6, 7, 8, and 9).

Analyzing the number 5,183:

- One's place: **3**
- Ten's place: **8**
- Hundred's place: **1**
- Thousand's place: **5**

Answer 82

Place weights: one's, twenty's, four hundred's, eight thousand's, . . .

one hundred fifty four   three hundred sixty four   four thousand seven hundred nine   one thousand one hundred forty
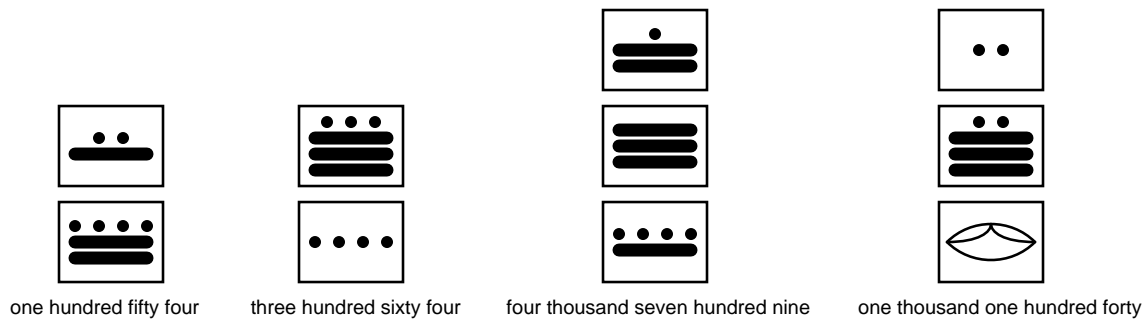
Answer 83

Five denary digits: one hundred thousand unique numbers. Six denary digits: one million unique numbers.

Three vigesimal "digits": eight thousand unique numbers. Four vigesimal "digits": one hundred sixty thousand unique numbers.

Four binary bits: sixteen unique numbers. Eight binary bits: two hundred fifty six unique numbers.

$$\text{Unique numbers } = b^n$$

Where,
$b = $ Radix of numeration system.
$n = $ Number of "places" given.

Answer 84

Twenty one bits.

Hint: the answer consists in solving this equation:

$$1300762 = 2^n$$

Where,
$n = $ The number of binary bits necessary to count up to 1,300,762.

Answer 85

In this system, with a "width" of 30 bits, we are able to represent one billion, seventy three million, seven hundred forty one thousand, eight hundred twenty four unique numbers. The greatest individual number which may be communicated is one less than this total.

Answer 86

The two states of an on/off circuit are equivalent to 0 and 1 ciphers in binary. Any medium in which on/off states may be physically represented is applicable to storing binary numbers. Optical disks (CD-ROM, DVD) are an excellent example of this, with laser-burned "pits" representing binary bits.

Answer 87

$$\text{\textit{Binary place-weights}}$$

$$1001.1011_2 = 9.6875_{10}$$

$$\frac{1}{2^3} \quad \frac{0}{2^2} \quad \frac{0}{2^1} \quad \frac{1}{2^0} \bullet \frac{1}{2^{-1}} \quad \frac{0}{2^{-2}} \quad \frac{1}{2^{-3}} \quad \frac{1}{2^{-4}}$$

$$\text{\textit{Octal place-weights}}$$

$$4027.3612_8 = 2071.471191406_{10}$$

$$\frac{4}{8^3} \quad \frac{0}{8^2} \quad \frac{2}{8^1} \quad \frac{7}{8^0} \bullet \frac{3}{8^{-1}} \quad \frac{6}{8^{-2}} \quad \frac{1}{8^{-3}} \quad \frac{2}{8^{-4}}$$

$$\text{\textit{Hexadecimal place-weights}}$$

$$C1A6.32B9_{16} = 49574.198135376_{10}$$

$$\frac{C}{16^3} \quad \frac{1}{16^2} \quad \frac{A}{16^1} \quad \frac{6}{16^0} \bullet \frac{3}{16^{-1}} \quad \frac{2}{16^{-2}} \quad \frac{B}{16^{-3}} \quad \frac{9}{16^{-4}}$$

Answer 88

- byte = 8 bits
- nybble = 4 bits
- word = *depends on the system*

The term "word," is often used to represent 16 bits, but it really depends on the particular system being spoken of. A binary "word" is more accurately defined as the default width of a binary bit grouping in a digital system.

Follow-up question: what binary grouping corresponds to a single hexadecimal character?

Answer 89

Morse Code is digital, being composed of only two types of characters, just like ASCII. Also, its purpose is to convey alphanumeric information, just like ASCII.

Answer 90

"Parity bits" are used as a primitive form of error detection. Communications equipment making use of parity for error detection may either be configured for "even parity" or for "odd parity". I'll let you research and explain how parity bits are used in the transmission of ASCII data.

Answer 91

"Floating-point" numbers are the binary equivalent of scientific notation: certain bits are used to represent the mantissa, another collection of bits represents the exponent, and (usually) there is a single bit representing sign. Unfortunately, there are several different "standards" for representing floating-point numbers.

Notes

Notes 1

Ask your students to hypothesize why binary is used in digital circuitry rather than base-ten (denary) numeration. I've revealed that it's easier to build circuits representing quantities in binary than any other numerical base, but why?

Notes 2

Ask your students to explain why subscripts are desirable when writing numbers in binary, or in any other "base" numeration system for that matter.

Notes 3

The most important part of this question, of course, is the *method* of translating binary into decimal. Don't tell your students how to do this, as there are plenty of good references to be found on the procedure. If a student cannot research and understand how to convert binary into decimal without your help, they aren't trying hard enough!

Notes 4

The most important part of this question, of course, is the *method* of translating decimal into binary. Don't tell your students how to do this, as there are plenty of good references to be found on the procedure. If a student cannot research and understand how to convert decimal into binary without your help, they aren't trying hard enough!

Notes 5

There are many references from which students may learn to perform these conversions. You assistance should be minimal, as these procedures are simple to comprehend and easy to find.

Notes 6

There are many references from which students may learn to perform these conversions. You assistance should be minimal, as these procedures are simple to comprehend and easy to find.

Notes 7

Lots of conversions to do here! I particularly like the "table" format shown here for practicing numeration system conversions, because it compresses a lot of practice into a small space on paper, and also because it allows students to use different methods of conversion. For example, in converting a decimal number to the other forms, a student might choose to convert first to binary, then from binary to octal and hex. Or, alternatively, a student may choose to convert from decimal into hex, then from hex into binary, then from binary to octal, the last two conversions being especially easy.

Notes 8

Although the root of this question is nothing more than binary-to-hexadecimal conversion, it also introduces students to the concept of controlling bit states in microcomputer ports by writing hex values. As such, this question is *very* practical!

In case students ask, let them know that a dollar sign prefix is sometimes used to denote a hexadecimal number. Other times, the prefix 0x is used (e.g., $F3 and 0xF3 mean the same thing).

Notes 9

Ask your students to explain the method of conversion used in each case. It is rather simple, but important to understand nonetheless. Ask your students how this method compares with the conversion of whole-number values into decimal form.

Notes 10

Lots of conversions to do here! I particularly like the "table" format shown here for practicing numeration system conversions, because it compresses a lot of practice into a small space on paper, and also because it allows students to use different methods of conversion. For example, in converting a decimal number to the other forms, a student might choose to convert first to binary, then from binary to octal and hex. Or, alternatively, a student may choose to convert from decimal into hex, then from hex into binary, then from binary to octal, the last two conversions being especially easy.

Notes 11

In order to familiarize students with these "strange" numeration systems, I like to begin each day of digital circuit instruction with counting practice. Students need to be *fluent* in these numeration systems by the time they are finished studying digital circuits!

One suggestion I give to students to help them see patterns in the count sequences is "pad" the numbers with leading zeroes so that all numbers have the same number of characters. For example, instead of writing "10" for the binary number two, write "00010". This way, the patterns of character cycling (especially binary, where each successively higher-valued bit has half the frequency of the one before it) become more evident to see.

Notes 12

Ask your students to brainstorm possible applications for rotary encoders. Where might we use such a device? Also, ask them to contrast the two encoder types (1 bit versus 3-bit) shown in the question. What applications might demand the 3-bit, versus only require a 1-bit encoder?

Notes 13

Ask your students what patterns they notice in the Gray code sequence, as compared to the binary count. What difference do they see between binary and Gray code, analyzing the bit transitions from one number to the next?

Notes 14

This is perhaps the most important reason for using Gray code in encoder marking, but it is not necessarily obvious *why* to the new student. I found that making a physical mock-up of a binary-coded wheel versus a Gray-coded wheel helped me better present this concept to students. Those students with better visualization/spatial relations skills will grasp this concept faster than the others, so you might want to solicit their help in explaining it to the rest of the class.

Notes 15

There are many textbook references for the conversion process between binary and Gray code. Let your students research how the conversions are done!

Notes 16

There are many textbook references for the conversion process between binary and Gray code. Let your students research how the conversions are done!

Notes 17

ASCII is arguably the *lingua franca* of the digital world. Despite its humble beginnings and Anglo-centric format, it is used worldwide in digital computer and telecommunication systems. Let your students know that every plain-text computer file is nothing more than a collection of ASCII codes, one code for each text character (including spaces).

Notes 18

This question provides students with practice using an ASCII reference table.

Notes 19

Discuss with your students the purpose of using BCD to represent decimal quantities. While not an efficient usage of bits, BCD certainly is convenient for representing decimal figures with discrete (0 or 1) logic states.

Notes 20

Nothing but straightforward conversions here!

Notes 21

In order to familiarize students with these "strange" numeration systems, I like to begin each day of digital circuit instruction with counting practice. Students need to be *fluent* in these numeration systems by the time they are finished studying digital circuits!

One suggestion I give to students to help them see patterns in the count sequences is "pad" the numbers with leading zeroes so that all numbers have the same number of characters. For example, instead of writing "10" for the binary number two, write "00010". This way, the patterns of character cycling (especially binary, where each successively higher-valued bit has half the frequency of the one before it) become more evident to see.

Notes 22

Ask your students to describe what differences exist between manually adding binary numbers and manually adding decimal numbers, if any.

Notes 23

Although this may seem like a trivial question, I've met electronics technicians who actually believed that the form of numeration affected the outcome of certain mathematical operations. In particular, I met one fellow who believed the number π was fundamentally different in binary form than it was in decimal form: that a binary "pi" was not the same quantity as a decimal "pi". I challenged his belief by applying some Socratic irony:

Me: How do you use a hand calculator to determine the circumference of a circle, given its diameter? For example, a circle with a diameter of 5 feet has a circumference of . . .

Him: By multiplying the diameter times "pi". 5 feet times "pi" is a little over 15 feet.

Me: Does a calculator give you the correct answer?

Him: Of course it does.

Me: Does an electronic calculator use decimal numbers, internally, to do math?

Him: No, it uses binary numbers, because its circuitry is made up of logic gates . . . *(long pause)* . . . Oh, now I see! If the type of number system mattered in doing math, digital computers and calculators would arrive at different answers for arithmetic problems than we would doing the math by hand!

Of course, those familiar with computer programming and numerical analysis understand that digital computers can introduce "artifacts" into computed results that are not mathematically correct. However, this is not due to their use of binary numeration so much as it is limited word-widths (leading to overflow conditions), algorithmic problems converting floating-point to integer and visa-versa, and such.

Notes 24

The principle of a "one's complement" is very, very simple. Don't give your students any hints at all concerning the technique for finding a one's complement. Rather, let them research it and present it to you on their own!

Be sure to discuss the follow-up question, concerning the one's complement of different-width binary numbers. There is a very important lesson to be learned here!

Notes 25

The point about word-length is extremely important. One cannot arrive at a definite two's complement for any number unless the word length is first known!

Notes 26

Students accustomed to checking their conversions with calculators may find difficulty with these examples, given the negative place weight! Two's complement notation may seem unusual at first, but it possesses decided advantages in binary arithmetic.

Notes 27

The most important concept in this question is that of *range*: what are the limits of the representable quantities, given a certain number of bits. Two's complement just makes the concept a bit more interesting.

Notes 28

Have your students do some of these problems on the board, in front of class for all to see. Ask students what happens to the left-most "carry" bit, if it exists in any of these problems. Ask them *why* we do what we do with that bit, when we would usually place it in our answer.

Notes 29

This question introduces students to the phenomenon of *overflow*. This is a very important principle to understand, because real computer systems must deal with this condition properly, so as not to output incorrect answers!

Notes 30

Later, this concept of overflow checking should be applied to a real circuit, with students designing logic gate arrays to detect the presence of overflow. First, though, they must learn to recognize its presence analytically.

Notes 31

In order to familiarize students with the standard logic gate types, I like to given them practice with identification and truth tables each day. Students need to be able to recognize these logic gate types at a glance, or else they will have difficulty analyzing circuits that use them.

Notes 32

Ask your students to describe what differences exist between manually adding binary numbers and manually adding decimal numbers, if any.

Notes 33

Have your students explain their design process to you, step by step. This circuit diagram is easy enough to discover in the pages of a textbook, so don't be surprised if students simply copy what they see without trying to understand how it works!

Notes 34

Have your students explain their design process to you, step by step. This circuit diagram is easy enough to discover in the pages of a textbook, so don't be surprised if students simply copy what they see without trying to understand how it works!

Deriving the two cascaded Ex-OR gates from the boolean expression is a bit tricky, but not impossible. Remind your students if necessary that the boolean equivalent for the Ex-OR function is $\overline{A}B + A\overline{B}$, and that the Ex-NOR function is $AB + \overline{A}\,\overline{B}$.

Notes 35

The "ripple" effect seen in simple binary adder circuits is not limited to adders! Some Gray-to-binary code converters and counter circuits also exhibit ripple, with the same deleterious effects.

Notes 36

This question is not really specific to adder/summer circuits, as it may first appear. The fundamental comparison being drawn in this question is between digital and analog. This is an important concept for students to grasp, as both have their roles in modern electronics. A common fallacy is that "digital is better" in all circumstances, but the truth is that both digital and analog have their respective strengths and limitations.

Notes 37

Be sure to ask students where they obtained their information. It is very easy to obtain datasheets online (over the internet), making it convenient to assign short research projects such as this.

Notes 38

Although the 74181 ALU is a somewhat dated IC (in fact, some versions are obsolete as of this writing – 2005), it stands as a simple example for students to learn from. A circuit such as this provides a good example of the power of integration, as opposed to constructing a similar logic function from individual gates (not to mention discrete transistors!).

The follow-up question brings up a point many students are confused on: the distinction between binary (numerical) and boolean (bitwise) operations. Binary is a place-weighted *numeration system*, used to symbolize real numbers using only two states per place. Boolean is a *number system* characterized by having only two possible values. Since both binary and boolean have something to do with two-valued quantities, many students believe the two to be interchangeable terms and concepts. However, they are not, and an investigation of the two operating modes of this ALU highlights the distinction.

Notes 39

A lot of arithmetic tricks existing in the decimal numeration system are applicable, with slight revision, in the binary numeration system as well. This is a popular one, and often used by shrewd computer programmers to execute fast multiply-by-two or divide-by-two operations when "conventional" multiplication commands take more time.

Notes 40

The answer pretty much says it all. The fact that the IC is an ALU is quite incidental. Some students may research the part number to get a better understanding of what is going on. That is fine, but my emphasis in this question is the schematic diagram convention(s) for multi-conductor busses.

Notes 41

The given answer does not provide enough detail to explain *why* TTL inputs tend to float high, so I recommend you display an internal TTL gate schematic for your students to analyze and comment on in class.

Notes 42

Ask your students to explain their answer based on an analysis of the internals of a CMOS gate, versus the internals of a TTL gate. Memorization is not good enough – students must grasp *why* these different logic families behave as they do.

Notes 43

Students think I'm a wizard by being able to troubleshoot their CMOS circuits just by waving my hand next to them. No, I'm just wise in the ways of common student error!

Notes 44

Many of the old 74xx and 74LSxx logic circuits are considered obsolete, but may still be found in a lot of operating equipment! It is not uncommon to have students mistakenly research the datasheets of a newer logic family such as 74HCxx which has different power supply requirements than traditional TTL. Be prepared to elaborate on the difference(s) between these IC families if and when your students encounter this confusion!

Notes 45

It is easy for students to overlook the limitations of a logic probe, and to forget what actually drives it to say "high" or "low" when measuring a logic level. This is why in low-speed circuits I prefer to use a good digital voltmeter rather than a logic probe to discern logic states. With a voltmeter, you can see exactly what the voltage level is, and determine whether or not the logic state is marginal.

Notes 46

Students may wonder how a logic pulser is able to override the output state of any gate it's connected to. The answer has to do with the brief time that a pulser operates, and also the pulser's low impedance (compared to the relatively high impedance of the overridden gate output).

Notes 47

The point I am trying to convey with this question is that forcing a gate's output high or low with a logic pulser tells us nothing about that gate. We use a pulser to override gate outputs in order to test the function of gates *receiving* that signal. In other words, we use a pulser to test gates "downstream" of where the pulser contacts the circuit.

Notes 48

Questions like this help students hone their troubleshooting skills by forcing them to think through the consequences of each possibility. This is an essential step in troubleshooting, and it requires a firm understanding of circuit function.

If a student suggests switch B failing open could cause the output to remain low, they are either misunderstanding the operation of an OR gate, or they are assuming switch A has also failed open (or something else happening that keeps input A low all the time). I find this kind of assumption frequently in students new to troubleshooting: assuming multiple faults. While multiple faults are not impossible, they are less likely than single faults. Therefore, the good technician *first* looks for single faults capable of accounting for all observed states before looking for (less likely) combinations of faults. This is a practical application of Occam's Razor.

Notes 49

Questions like this help students hone their troubleshooting skills by forcing them to think through the consequences of each possibility. This is an essential step in troubleshooting, and it requires a firm understanding of circuit function.

Notes 50

Ask your students to explain what logic state is *supposed* to be at that point in the circuit, and what logic state they *suspect* might be there that could account for the aberrant output. Also discuss why this particular choice of switch settings is the best for a first test.

If students do not immediately grasp why the switches should be set as the answer indicates, pose the following scenario. Suppose they were asked to troubleshoot a simple light bulb circuit using only a voltmeter. The problem is, the light bulb does not come on when the switch is closed. Would it be best to proceed with their voltage measurements with the switch on or off? It should be easy to understand that having the switch in the off position would only interfere with the diagnosis, and that turning the switch on is the best way to reveal the fault (so that one could use the voltmeter to see where voltage is not present, but should be). Likewise, it is wise to set up this faulted logic circuit in such a way that the output ought to be doing something it isn't. This way, one may easily compare logic states as they are versus as they should be, and from there determine what type of fault could be causing the problem.

Notes 51

Discuss fault options with your students, asking them to explain *why* each proposed fault would result in the solenoid not energizing.

Notes 52

Questions like this help students hone their troubleshooting skills by forcing them to think through the consequences of each possibility. This is an essential step in troubleshooting, and it requires a firm understanding of circuit function.

Notes 53

Questions like this help students hone their troubleshooting skills by forcing them to think through the consequences of each possibility. This is an essential step in troubleshooting, and it requires a firm understanding of circuit function.

Notes 54

Questions like this help students hone their troubleshooting skills by forcing them to think through the consequences of each possibility. This is an essential step in troubleshooting, and it requires a firm understanding of circuit function.

Notes 55

Questions like this help students hone their troubleshooting skills by forcing them to think through the consequences of each possibility. This is an essential step in troubleshooting, and it requires a firm understanding of circuit function.

Notes 56

I strongly recommend having students build their logic circuits with CMOS chips rather than TTL, because of the less stringent power supply requirements of CMOS. I also recommend drawing a combinational circuit using four gates, because this is the common number of two-input gates found on 14-pin DIP logic chips.

Notes 57

It should be noted that the input states in this circuit are defined by the voltage levels, not by the contact status. In other words, a closed contact equals a "low" (0) logic state.

I strongly recommend having students build their logic circuits with CMOS chips rather than TTL, because of the less stringent power supply requirements of CMOS. I also recommend drawing a combinational circuit using four gates, because this is the common number of two-input gates found on 14-pin DIP logic chips.

Notes 58

It should be noted that the input states in this circuit are defined by the voltage levels, not by the contact status. In other words, a closed contact equals a "low" (0) logic state.

I strongly recommend having students build their logic circuits with CMOS chips rather than TTL, because of the less stringent power supply requirements of CMOS. I also recommend drawing a combinational circuit using four gates, because this is the common number of two-input gates found on 14-pin DIP logic chips.

Notes 59

In this activity, students are asked to figure out the details of configuring the ADC: what power supply voltage to use, what resistor values, etc. The best source for this information is the ADC's datasheet!

For those students who have trouble figuring out how to calculate resolution, I recommend the following formula:

$$V_{resolution} = \frac{V_{range}}{2^n - 1}$$

Where,

V_{range} = "Span" of analog voltage input (how many volts of range it has from 00000000 to 11111111)

n = Number of output bits for the ADC

Notes 60

The purpose of this assessment rubric is to act as a sort of "contract" between you (the instructor) and your student. This way, the expectations are all clearly known in advance, which goes a long way toward disarming problems later when it is time to grade.

Notes 61

Note that the circuit in figure 4 is a "trick:" two of the resistors contribute absolutely nothing to R_{AB}! Be sure to discuss why this is with your students.

Discuss with your students how they approached each of these problems, and let the entire class participate in the reasoning process. The point of this question, like most of the questions in the Socratic Electronics project, is not merely to obtain the correct answers, but to stimulate understanding of *how* to solve problems such as these.

Notes 62

Discuss with your students what a good procedure might be for calculating the unknown values in this problem, and also how they might check their work.

Students often have difficulty formulating a method of solution: determining what steps to take to get from the given conditions to a final answer. While it is helpful at first for you (the instructor) to show them, it is bad for you to show them too often, lest they stop thinking for themselves and merely follow your lead. A teaching technique I have found very helpful is to have students come up to the board (alone or in teams) in front of class to write their problem-solving strategies for all the others to see. They don't have to actually do the math, but rather outline the steps they would take, in the order they would take them.

By having students outline their problem-solving strategies, everyone gets an opportunity to see multiple methods of solution, and you (the instructor) get to see how (and if!) your students are thinking. An especially good point to emphasize in these "open thinking" activities is how to check your work to see if any mistakes were made.

Notes 63

A noteworthy feature of this circuit's schematic is how the power supply connections are shown. Unlike many of my schematic diagrams, I do not show a "battery" symbol here for a voltage source. Instead, I show power supply "rail" symbols (flat line and a ground symbol). Let your students know that this is very common symbolism in modern schematics, and that is merely saves having to draw lines to a voltage source symbol (as well as the source symbol itself).

Discuss with your students what a good procedure might be for calculating the unknown values in this problem, and also how they might check their work.

Students often have difficulty formulating a method of solution: determining what steps to take to get from the given conditions to a final answer. While it is helpful at first for you (the instructor) to show them, it is bad for you to show them too often, lest they stop thinking for themselves and merely follow your lead. A teaching technique I have found very helpful is to have students come up to the board (alone or in teams) in front of class to write their problem-solving strategies for all the others to see. They don't have to actually do the math, but rather outline the steps they would take, in the order they would take them.

By having students outline their problem-solving strategies, everyone gets an opportunity to see multiple methods of solution, and you (the instructor) get to see how (and if!) your students are thinking. An especially good point to emphasize in these "open thinking" activities is how to check your work to see if any mistakes were made.

Notes 64

Discuss with your students what a good procedure might be for calculating the unknown values in this problem, and also how they might check their work.

Students often have difficulty formulating a method of solution: determining what steps to take to get from the given conditions to a final answer. While it is helpful at first for you (the instructor) to show them, it is bad for you to show them too often, lest they stop thinking for themselves and merely follow your lead. A teaching technique I have found very helpful is to have students come up to the board (alone or in teams) in front of class to write their problem-solving strategies for all the others to see. They don't have to actually do the math, but rather outline the steps they would take, in the order they would take them.

By having students outline their problem-solving strategies, everyone gets an opportunity to see multiple methods of solution, and you (the instructor) get to see how (and if!) your students are thinking. An especially good point to emphasize in these "open thinking" activities is how to check your work to see if any mistakes were made.

Notes 65

Discuss with your students what a good procedure might be for calculating the unknown values in this problem, and also how they might check their work.

Notes 66

Kirchhoff's Voltage Law (KVL) is very easily explored in real life with a set of batteries and "jumper wire" connections. Encourage your students to build battery circuits like the one shown in this question, to be able to see the results for themselves!

One really nice feature of digital multimeters (DMMs) is the ability to register negative as well as positive quantities. This feature is very useful when teaching Kirchhoff's Laws, with the *algebraic* (sign-dependent) summation of voltages and currents.

Notes 67

This question helps to familiarize students with the concept of "ground" as the default reference point for taking voltage measurements, as well as being an application of Kirchhoff's Voltage Law.

Notes 68

In your discussion, be sure to explore more than one "loop" when using KVL. Not only does this demonstrate the arbitrary nature of your loop choice, but it also serves as a double-check for your work!

It is not necessary to know anything about series-parallel or even parallel circuits in order to solve the R_4's voltage – all one needs to know is how to use Kirchhoff's Voltage Law.

Notes 69

Ask your students to explain how they obtained their answers. There is more than one correct way to answer this question!

Notes 70

Discuss with your students multiple techniques of solving for these voltages, asking them first for their solution strategies.

Notes 71

Discuss with your students multiple techniques of solving for these voltages, asking them first for their solution strategies.

Notes 72

I want students to see that there are two different ways of approaching a problem such as this: with *scalar* math and with *complex number* math. If students have access to calculators that can do complex-number arithmetic, the "complex" approach is actually simpler for series-parallel combination circuits, and it yields richer (more informative) results.

Ask your students to determine which of the approaches most resembles DC circuit calculations. Incidentally, this is why I tend to prefer complex-number AC circuit calculations over scalar calculations: because of the conceptual continuity between AC and DC. When you use complex numbers to represent AC voltages, currents, and impedances, almost all the rules of DC circuits still apply. The big exception, of course, is calculations involving *power*.

Notes 73

I want students to see that there are two different ways of approaching a problem such as this: with *scalar* math and with *complex number* math. If students have access to calculators that can do complex-number arithmetic, the "complex" approach is actually simpler for series-parallel combination circuits, and it yields richer (more informative) results.

Ask your students to determine which of the approaches most resembles DC circuit calculations. Incidentally, this is why I tend to prefer complex-number AC circuit calculations over scalar calculations: because of the conceptual continuity between AC and DC. When you use complex numbers to represent AC voltages, currents, and impedances, almost all the rules of DC circuits still apply. The big exception, of course, is calculations involving *power*.

Notes 74

Some students may wonder why every side of the triangle is represented by a Y term, rather than Y for the hypotenuse, G for the adjacent, and B for the opposite. If students ask about this, remind them that conductance (G) and susceptance (B) are simple two different *types* of admittances (Y), just as resistance (R) and reactance (X) are simply two different types of impedances (Z).

Notes 75

Some electronics textbooks (and courses) tend to emphasize scalar impedance calculations, while others emphasize complex number calculations. While complex number calculations provide more informative results (a phase shift given in *every* variable!) and exhibit conceptual continuity with DC circuit analysis (same rules, similar formulae), the scalar approach lends itself better to conditions where students do not have access to calculators capable of performing complex number arithmetic. Yes, of course, you can do complex number arithmetic without a powerful calculator, but it's a *lot* more tedious and prone to errors than calculating with admittances, susceptances, and conductances (primarily because the phase shift angle is omitted for each of the variables).

Notes 76

Too many students blindly use impedance and voltage triangles without really understand what they are and why they work. These same students will have no idea how to approach a problem like this. Work with them to help them understand!

Notes 77

This is a very practical application of resistor-capacitor (RC) circuits: to introduce a phase shift to an AC signal. Examples of where a circuit such as this may be used include oscillators (to introduce phase shift into a feedback network for a total phase shift of 360°) and thyristor firing control circuits (phase-shifting the triggering voltage in relation to the source voltage).

Notes 78

Phase-shifting circuits are very useful, and important to understand. They are particularly important in some types of oscillator circuits, which rely on RC networks such as this to provide certain phase shifts to sustain oscillation.

Notes 79

Discuss with your students what a good procedure might be for calculating the unknown values in this problem, and also how they might check their work.

Students often have difficulty formulating a method of solution: determining what steps to take to get from the given conditions to a final answer. While it is helpful at first for you (the instructor) to show them, it is bad for you to show them too often, lest they stop thinking for themselves and merely follow your lead. A teaching technique I have found very helpful is to have students come up to the board (alone or in teams) in front of class to write their problem-solving strategies for all the others to see. They don't have to actually do the math, but rather outline the steps they would take, in the order they would take them.

By having students outline their problem-solving strategies, everyone gets an opportunity to see multiple methods of solution, and you (the instructor) get to see how (and if!) your students are thinking. An especially good point to emphasize in these "open thinking" activities is how to check your work to see if any mistakes were made.

Discuss with your students what a good procedure might be for calculating the unknown values in this problem, and also how they might check their work.

Students often have difficulty formulating a method of solution: determining what steps to take to get from the given conditions to a final answer. While it is helpful at first for you (the instructor) to show them, it is bad for you to show them too often, lest they stop thinking for themselves and merely follow your lead. A teaching technique I have found very helpful is to have students come up to the board (alone or in teams) in front of class to write their problem-solving strategies for all the others to see. They don't have to actually do the math, but rather outline the steps they would take, in the order they would take them.

By having students <u>outline their problem-solving strategies</u>, everyone gets an opportunity to see multiple methods of solution, and you (the instructor) get to see how (and if!) your students are thinking. An especially good point to emphasize in these "open thinking" activities is how to check your work to see if any mistakes were made.

The "base" of our numeration system is something people usually don't think much about – it is simply taken for granted. The purpose of this question is to help students realize what numerical symbols actually mean, in preparation for understanding other systems of numeration.

Although no one counts in "Mayan" anymore, this question is still relevant because it gets students thinking outside their accustomed numeration system. Also, it has some cultural value, showing them that not everyone in the world counts the same way they do!

Ask your students which numeration system of the three given is more efficient in representing quantities with the smallest number of places. Then, ask them to explain *why* this is.

This question is a test of whether or not students know how to algebraically solve for variable exponents. I'm not going to suggest how this is done (lest I rob a student of the learning experience that comes from researching the answer on their own), but I will say that it is a *very* useful algebraic technique, once mastered.

If students still haven't found a solution after doing their research, suggest they try to solve a simpler problem:

How many *decimal digits* are needed to count up to the number one million three hundred thousand seven hundred sixty two?

This question is trivial to answer (7 decimal digits), since we're all familiar with decimal numeration. However, the real learning takes place when students write a mathematical expression for solving this problem, similar to the one written in the answer for the binary problem. Once they have that expression written, ask them what algebraic techniques could be used to solve for the exponent's value.

The purpose of this question is to allow students to consider an electric circuit that communicates digital quantities from one location to another, rather than abstractly discuss numeration systems. It also provides a more practical context in which to understand maximum count in a numeration system.

Ask your students to think of different media or physical quantities which may represent binary information, especially those related to electric/electronic circuits.

Many students will not realize initially that it is possible to represent non-whole numbers in binary, octal, or hexadecimal. Really, though, the concept is identical to the representation of non-whole numbers in decimal form. The ability of your students to grasp non-whole numbers in these other numeration systems indicates their grasp of place-weighted systems in general. If a student truly comprehends how place-weighting works, they will have no trouble understanding digits to the left *or* the right of the radix point in any numeration system. If a student does not understand how digits to the right of the decimal point are interpreted in other numeration systems, then they need to spend more time reviewing what decimal numbers mean.

It's not that the concept is so hard to understand, so much as it is our familiarity with the decimal (base-ten) numeration system. We grow so accustomed to one way of representing numbers that we don't realize what the symbols actually *mean*, or that there may be alternative methods of representing quantities.

Definitions taken from *The New Hacker's Dictionary*, available on internet terminals everywhere.

An interesting point to bring up to students about Morse Code is that it is *self-compressing*. Note how different Morse characters possess different "bit" lengths, whereas ASCII characters are all 7 bits each (or 8 bits for Extended ASCII). This makes Morse a more efficient code than ASCII, from the perspective of bit economy!

Ask your students what ramifications this "self-compressing" aspect of Morse Code would have if we were to choose it over ASCII for sending alphanumeric characters over digital communications lines, or store alphanumeric characters in some form of digital memory media.

The concept of parity is not very complex. It should be well within the reach of students to research on their own and report their findings to the class as a whole.

Ask your students why computer systems would have need for floating point numbers. What's wrong with the standard forms of binary numbers that we've explored thus far?