IP subnetting is an elementary topic that is essential for every IP network engineer to understand. Many individuals have a difficult time grasping the subject and traditionally are left completely baffled that they just give up. They search for short cuts or try to memorize charts and tables, any simple method to achieve their objective. When in fact what they should be doing is learning how Binary math can be their friend when it comes to IP subnetting. Remember back in grade school when you first started learning division. Your math teacher showed you this drawn out “long” division method for dividing two numbers. But once you understood the series of steps involved the problems become much easier to solve. You were even able to tackle more complex and arbitrarily large numbers by following these simple steps. This is what Binary math does for us when dealing with IP subnetting. By following a series of seven simple steps we can become extremely proficient at even the most challenging subnetting questions.

Before we get started lets first go over a few terms and definitions we will be using as you read through this document. We have all heard, or at least I think we have heard of the following terms, network address, broadcast address, subnet, prefix length and subnet mask. Furthermore, it is assumed that the reader already has an understanding of the basic IP addressing terms and their definitions. The list below introduces a few new terms used in Binary math as well as some not so familiar ones.

<table>
<thead>
<tr>
<th>Term</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>NA</td>
<td>abbreviation for the network address or the all “zeros” address</td>
</tr>
<tr>
<td>BA</td>
<td>abbreviation for the broadcast address or the all “ones” address</td>
</tr>
<tr>
<td>NM</td>
<td>abbreviation for the net mask, which is a 32-bit combination used to describe which portion of an address belongs to the network and which part belongs to the host</td>
</tr>
</tbody>
</table>
SNM abbreviation for the subnet mask, which is a 32-bit combination used to describe which portion of an address belongs to the subnet and which part belongs to the host.

IM abbreviation for the inverse mask

FVH abbreviation for the first valid host within a network or subnetwork

LVH abbreviation for the last valid host within a network or subnetwork

AND a binary logic function used in binary circuits to compare two binary numbers, resulting in a 1 only when both numbers are 1, otherwise result is 0

XOR a binary logic function used in binary circuits to compare two binary numbers, resulting in a 1 only when the two numbers are different, otherwise result is 0

OR a binary logic function used in binary circuits to compare two binary numbers, resulting in a 1 when any number is a 1, otherwise result is 0

Below are the truth tables for the binary logic functions briefly described above.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
The types of subnetting questions you encounter will most likely vary in both the information supplied and the information requested. You might be asked to find the network or broadcast address to which a host address belongs. Or to find the first or last valid host on a given network. Or maybe still one that asks a combination of these two types, plus the number of subnets or hosts. Regardless of the type of question, inorder to find the answers is going to involve math, both Binary and Decimal. I am going to show you how you can find the answer to virtually any type of question by following seven simple steps. They may be somewhat long and drawn out, but just as you learned long division you can learn these as well. Once you master these seven steps you will be able to tackle the most challenging subnetting questions and survive. Ready to get started?

Lets use for our example, a type of question most often seen on exams.

Question: What valid host range is the IP address 172.16.233.59/21 a part of?

Step 1.
Write down what you know from the information given in the question. We know the IP address, the default classful network, the prefix length, the number of network bits, the number of subnet bits, and the number of host bits. Do you see this?

IP address: 172.16.233.59
Default Classful network: Class B
prefix length: /21
network bits: 16
subnet bits: 21 – 16 = 5
host bits: 32 – 21 = 11
Now for this particular question we do not need all of this information. However, it is good practice to include everything you know, because you may need it in later steps depending on the type of question you have been asked.

Step 2.
Convert the prefix length to a subnet mask (SNM). Recall the default Classful network was a Class B address, so we have 16 bits or 255.255.0.0. Since we have 5 additional bits from the third octet used for the subnet portion, the remaining 3 bits belong to the host portion. Now here is where your powers of 2 kick in. Recall that \(2^3 = 8\) and \(256 – 8 = 248\).

We now have the SNM: 255.255.248.0.

Step 3.
This is where we get down and dirty with the binary math. Now some of you may be familiar with the process of taking the subnet mask and anywhere you see 255, write down the address and where you see zero, write down zero. But have you ever asked yourself why this works? It is due to the binary AND function we apply in this step. Recall the AND function truth table from page 2. I will show all octets so you can understand the process and why the above short cut works.

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>172.016.233.059</td>
<td>10101100.00010000.11101001.00111011</td>
</tr>
<tr>
<td>255.255.248.000</td>
<td>11111111.11111111.11111000.00000000</td>
</tr>
<tr>
<td>172.016.232.000</td>
<td>10101100.00010000.11101000.00000000</td>
</tr>
</tbody>
</table>

We now have the NA: 172.16.232.0, this is the lower end of the IP address range.

Step 4.
We convert the subnet mask to an inverse mask by using the binary XOR function and compare
the subnet mask with the all “ones” address. Some of you may recall the short cut whereby you simply subtract in decimal the subnet mask from the all “ones” address. Again, the reason the short cut works, is due to the binary XOR function. Recall the XOR function truth table from page 2, we again show the decimal and binary representation.

\[
\begin{align*}
255.255.255.255 & \quad 11111111.11111111.11111111.11111111 \\
255.255.248.000 & \quad 11111111.11111111.11111000.00000000 \\
000.000.007.255 & \quad 00000000.00000000.00000111.11111111
\end{align*}
\]

We now have the IM: 0.0.7.255

Step 5.
We now add the inverse mask (IM) to the network address (NA) found in Step 4 to find the broadcast address (BA). Some may recognize the short cut method here, where we add in decimal the inverse mask (IM) to the network address (NA). Again, the reason the short cut works is due to the binary OR function we apply in this step. Recall the OR function truth table from page 2, we again show the decimal and binary representation.

\[
\begin{align*}
172.016.232.000 & \quad 10101100.00010000.11101000.00000000 \\
000.000.007.255 & \quad 00000000.00000000.00000111.11111111 \\
172.016.239.255 & \quad 10101100.00010000.11101111.11111111
\end{align*}
\]

We now have the BA: 172.16.239.255, this is the upper end of the IP address range.

Step 6.
We calculate the first valid host (FVH) by adding one to the network address (NA). No more binary math from here on out. Just simple decimal math \( \text{NA} + 1 = \text{FVH} \).
We now have the FVH: 172.16.232.1

Step 7.
We calculate the last valid host (LVH) by subtracting one from the broadcast address (BA). Just simple decimal math BA – 1 = LVH.

We now have the LVH: 172.16.239.254

Referring back to the original question and taking the answers from Steps 6 and 7, we have the following:

Question: What valid host range is the IP address 172.16.233.59/21 a part of?

Answer: Range of valid hosts = FVH to LVH = 172.16.232.1 to 172.16.239.254

Although the answer to this question was not found until the end, we may not have to perform each and every step to solve each question. Again, depending on what the question asks for will depend on how many steps we must go through. But one thing is certain, we now have a very simple seven step process to solve any IP subnetting question. Lets summarize everything we knew before we started and all that we found during the seven step process.

IP address: 172.16.233.59
Prefix length: /21
# of network bits: 16
# of subnet bits: 5
# of host bits: 11
# of subnets: $2^5 = 32$
# of hosts per subnet: $2^{11} - 2 = 2046$
SNM: 255.255.248.0
IM: 0.0.7.255
NA: 172.16.232.0
FVH: 172.16.232.1
LVH: 172.16.239.254
BA: 172.16.239.255

Range of IP addresses: NA to BA = 172.16.232.0 to 172.16.239.255
Range of valid hosts: FVH to LVH = 172.16.232.1 to 172.16.239.254

In conclusion, IP subnetting is a very important and foundational topic requiring every IP network engineer to master. It is no wonder you are taught about its intricacies early in your quest for certification. It can seem daunting and at times overwhelming, but be rest assured, it can be learned. I hope you found this document easy to read and understand. More importantly, that you found it informative and worth reading. I wrote this document with the intention to take the fear out of learning binary math, while making IP subnetting simple and easy to understand. I hope you acquired some valuable knowledge and may you walk away with a renewed level of confidence in your IP subnetting abilities.