

Design and Implementation of a VLSM Simulator

Hussein Abdul-Razzaq Lafta
Department of Computer Engineering
Baghdad University
Baghdad, Iraq
Email: enghabdul@coeng.uobaghdad.edu.iq

Ammar Osamah Hoori
Department of Computer Engineering
Baghdad University
Baghdad, Iraq
Email: ammar.osama@coeng.uobaghdad.edu.iq

Abstract

Variable-Length Subnet Masks (VLSM), often referred to as "subnetting a subnet", is used to maximize addressing efficiency. The network administrator is able to use a long mask on networks with few hosts, and a short mask on subnets with many hosts. This addressing scheme allows growth and does not involve wasting addresses. VLSM gives a way of subnetting a network with minimal losses of IP addresses for a specific range.

Unfortunately, the network administrator has to perform several mathematical steps (or use charts) to get the required results from VLSM.

In this paper, a simple graph simulator is proposed (using Visual Basic 6.0 Language) to perform all the required mathematical steps and to display the obtained required information (the subnet ID, broadcast ID, usable addresses for sub networks and others). The simulator also includes the ability to draw a suggested network topology that matches the entries.

The implementation of the simulation required only very few entries (IP, prefix and number of subnets). This simulator is useful for students, instructors, and network engineers to analysis and design a VLSM network by providing all required information in simple, fast and easy steps. Moreover, the software draws a full detailed suggested network topology which is considered a helpful tool for the network administrator that he should have.

Keywords: VLSM, IP Addressing, Subnetting, Maximize Addressing Efficiency.

الخلاصة:

الشبكات الفرعية ذات القناع المتغير (variable length subnet mask) عادة تشير بها الى تجزئة الشبكات الفرعية، وهي تستخدم لزيادة كفاءة العنوان. بحيث يتمكن مدير الشبكة من استعمال قناع طويل من الشبكات ذات الاجهزة القليلة وقناع قصير مع الشبكات ذات الاجهزة العديدة. طريقة العنوان هذه تسمح بالنمو للعناوين ولا تتضمن خسارة في العناوين. الشبكة الفرعية ذات القناع المتغير تعطي طريقة لتجزئة الشبكات بأقل خسائر للعناوين لمدى معين من العناوين. لسوء الحظ، مدير الشبكة يتوجب عليه القيام بعدة عمليات رياضية (أو استعمال مخططات معينة) للحصول على النتائج المطلوبة من الشبكات الفرعية ذات القناع المتغير.

في هذه الورقة البحثية، نقترح محاكي بسيط مع قابلية رسم للشبكة (باستعمال لغة برمجة Visual Basic 6.0) للقيام بكل العمليات الرياضية المطلوبة وعرض النتائج المستخلصة (عنوان الشبكات الفرعية وعنوان البث لكل شبكة فرعية و بداية ونهاية عناوين الاجهزة المستخدمة في كل شبكة فرعية وغيرها). المحاكي ايضاً يتمتع بالقدرة على رسم مخطط لشبكة مقترحة والتي توافق المدخلات.

تصميم المحاكي يتطلب عدد قليل من المدخلات مثل (رقم الشبكة و ملحق الشبكة وعدد الشبكات الفرعية). يعتبر هذا البرنامج مهم للطلبة، للمدرسين ولمهندسي شبكات الحاسبات لتحليل وتصميم الشبكة الفرعية ذات القناع المتغير عن طريق توفير كل المعلومات المطلوبة بخطوات بسيطة وسريعة وسهلة. إضافة لذلك، البرنامج يرسم طوبوغرافية الشبكة بكافة التفاصيل والتي تعد أداة مفيدة لمدير الشبكة يتوجب عليه امتلاكها.

Introduction

Classical IP Addressing Scheme

An IP address consists of 32 bits of information. These bits are divided into four bytes; it is structured or hierarchical address, the advantage of this scheme is that it can handle a large number of addresses, namely 4.3 billion. Subdividing an IP address into a network and node address is determined by the class designation of one's network. Fig. 1 summarizes the three classes of networks. [4]

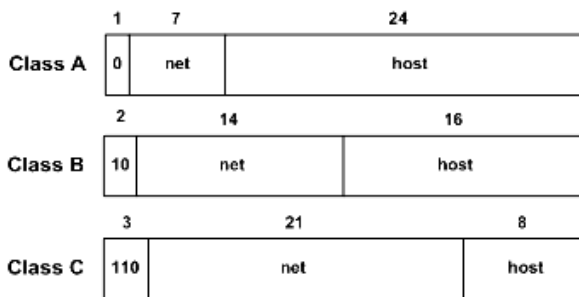


Fig. 1: Summary of the three network classes

The designers of the Internet decided to create classes of networks based on network size. For the small number of networks possessing a very large number of nodes, they created the rank Class A network. At the other extreme is the Class C network, which is reserved for the numerous networks with a small number of nodes. The class distinction for networks between very large and very small is predictably called the Class B network. Every machine on the same network shares that network address as part of its IP address. The node address is assigned to, and uniquely identifies, each machine on a network. This number can also be referred to as a host address. Table 1 shows the more details about three classes of networks.[4]

Fixed-Length Subnet Masks (Subnetting)

As discussed in previous section, the IP address space features a two-tier hierarchy in which each address consists of a network address and a host address within its 32-bit structure. Such flatness distinctly limits scalability in a number of ways. Perhaps the most confining limitation is that the address space assumes that all networks fit into one of just three different sizes of networks small, medium, and extremely large.

For example each class B network address contained 16 bits in the host portion, it controlled 65,534 addresses. (Remember, 2 addresses were reserved for the network and broadcast addresses).

Only the largest organizations and governments could ever hope to use all 65,000 addresses. The IP address space was tremendously wasted.

IP address space was depleting rapidly, for more efficient use of IPv4 address space, creating a third tier for identifying subnetwork addresses is a relatively straightforward concept that involves "borrowing" bits from the host portion of the address. These bits are used to create subnetwork addresses as an extension of the network address. In other words, smaller networks can be created and uniquely addressed from larger networks and network address spaces.

Subnetting a class A address space would have the tremendous benefit of making more efficient use of the available pool of addresses. Subnetting a class C network address space with just 254 total available host addresses makes quick aware of the finite nature of those addresses. Logically, the more borrowed bits from the host field to create subnets, the fewer bits that remain for the identification of hosts. Fig. 2 shows the format addresses when subnetting. [3]

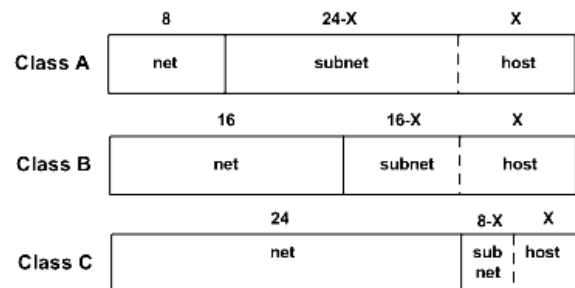


Fig. 2: Address formats when subnetting is used [6]

The Subnet Mask

A subnet mask is a 32-bit binary number; a subnet mask is structurally similar to an IP address. However, a subnet mask does serve an important function: It is used to tell end systems (including routers and hosts in the LAN) how many bits of the IP address's host field have been borrowed to identify the subnet. The bits in the mask that identify the network address, as well as the subnet address, are set to 1s. The remaining bits, which are used for host addresses within each subnet, are set to 0s.

Table 2 shows the dotted-decimal and dotted-binary forms of default subnet masks for class A, B, and C. Table 3 shows the dotted-decimal and dotted-binary forms of subnet masks

that are permissible when subnetting a class C address. [3]

Extended Network Prefix

It is important to note that the borrowed bits are always the leftmost bits in the host field. Thus, the subnet address is numerically contiguous with the network address. Together, they form the extended network prefix. The remaining bits are used for host identification. [3]

The extent of the increase in network bits depends on the number of required subnets and the number of required hosts on each subnet. Table 4 and Table 5 show Prefix Length and possible ways of subnetting class B and C networks. [2]

Variable-Length Subnet Masks

Subnetting, in general, was designed to enable more efficient use of address space by permitting class-based network address blocks to be subdivided into smaller address blocks. Yet the way subnetting was originally implemented was far from efficient. Source of waste and inefficiency with FLSM, There was one size of mask for all subnets. Implementing FLSM actually wastes IP addresses. [3]

For example, the serial link between two routers, as shown in Fig. 3, shares the same network to talk. Two IP numbers are required, one for each serial interface, unfortunately, eight-bit subnet mask are available (i.e., 255.255.255.0), so 252 IP addresses are wasted of the 254 available numbers on the subnet. One possible solution to this dilemma is to use Variable-Length Subnet Masks (VLSMs).

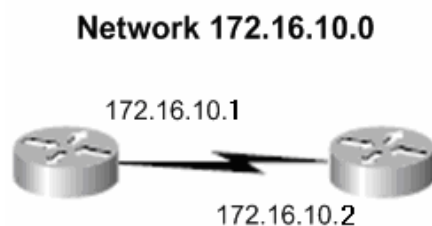


Fig. 3: IP address example

As the name suggests, with Variable-Length Subnet Masks, different subnet masks for different subnets are available. So, for the serial link in this example Fig. 3, the network address is 172.16.10.0/30 with subnet mask of 255.255.255.252. by calculation of IP addresses in this network one can see that only two host bits are considered, as shown in Table 6. [3]

Therefore, this subnet mask will give only two host IPs ($2^2 - 2 = 2$), which is exactly what is required for this serial link. [5]

Implementing VlsM Networks

To create VLSMs quickly and efficiently, there is needs to understand how block sizes and charts work together to create the VLSM masks. Table 7 shows the block sizes used when creating VLSMs with Class C networks. For example, if 25 hosts' subnet is required, then it belongs to a block size of 32. Also, if 11 hosts' subnet is required, then it belongs to a block size of 16. Moreover, if 40 hosts' subnet is required, then it belongs to a block size of 64. [4]

The next step in creating a VLSM network is by using the VLSM worksheet or chart. The reason of using this table and chart are to prevent overlap networks.

Visualizing Subnets Using a VlsM Chart

The VLSM chart is the method used to visualize the breakdown of subnets and addresses into smaller sizes. By shading or coloring in the boxes one can easily break up subnets without overlapping addresses. Each sub-subnet can be adjusted to the correct size needed. [1]

Example

A company has IP address range of 192.168.16.0/24 and this company wants to create four subnets (20, 60, 120, 2 hosts/subnet) as shown in Fig. 4.

To solve such issue, a VLSM method should starts with the largest groups (hosts/ subnet), i.e. decently (120, 60, 20, 2). Then by using the VLSM chart (shown in Fig. 5) one can distribute the required block size of each subnet into the right size by shading the required area in chart. Shading an area reserves it from another subnet placement. After distributing all subnets the resulted network will be as shown in Fig. 4.

Software Implementation Of The VlsM Simulator:

This section presents the software implementation of the VLSM Simulator that had been built from mathematical operation which discussed in previous sections. VLSM Simulator contains more than one window to display its operation; this section presents and explains the work of VLSM Simulator's windows and the relationship between these windows.

The Main Window of the VLSM Simulator:

Fig. 6 shows the main window of the VLSM Simulator, which consists of six parts,

Part1:- For entering the desired IP address to be analyzed, this part will determine the following:-

- 1- Default network prefix.
- 2- Network IP address that the desired IP address belongs to.
- 3- Broadcast IP address of this network.

Part2:- For displaying some attributes on the desired IP address, these attributes include:-

- 1- Class of the IP address (A, B or C).
- 2- Type of the IP address in the network (Host, Network or Broadcast IP address).
- 3- Category of the IP addresses (private or public IP address).
- 4- The usable host range before sub netting.

Part3:- For entering the needed number of subnets (default subnet equal to one), and for entering the number of hosts at each subnet.

Part4:- For displaying the calculated sub netting information, which includes:-

- 1- The lost IP addresses duo to sub netting (network IP address and broadcast IP address at each subnet).
- 2- The used IP addresses after sub netting (total reserved IP addresses for all subnets).
- 3- The number of remaining (not used) IP addresses.

Part5:- For generating VLSM topology table and drawing a suggested topology map, which includes each subnet and subnet's IP address ranges.

Part6:- For displaying the topology table which includes the required information needed by the network administrator to build the network.

Entering the IP Address and Generating VLSM

When the user enters the desired IP address and prefix of the network, the software will create the network and Broadcast IP addresses which the desired IP address belongs to. When he enters the number of subnets, the software will create a number of boxes equals to number of subnets. The next step is that each box must be filled with number of host required in each subnet.

The "Generate VLSM" button will be activated after entering the above parameters. Clicking this button, resulting in displaying a table that includes the important parameters about each subnet as shown in Fig. 7 and the flow chart is shown in Fig. 10 . These parameters are network name, Net IP, start IP, End IP, BC IP, prefix, Total hosts, used hosts and remaining hosts.

Draw Topology

One can click on "Draw Topology" button in the main window to draw and display the suggested topology with the least number of routers.

Each router in the suggested topology has at most two fast Ethernet ports and two serial ports. Note that detailed IP information of the network can be enabled or disabled by clicking on check box on the top of the "Draw Topology" window named as "show all networks IPs" command shown in Figs. (8.a) and (8.b).

Survey Questions to Test the Simulator

The simulator was tested by a group of volunteers those were students and were well-trained on VLSM sub netting. Each student was asked to solve four different questions by hand, and then resolve the same questions using the VLSM Simulator. Table 8 shows the required time (in minutes) by each student to answer each question once by hand and then by simulator. Fig. 9 shows a chart which includes the average time by hand compared to the average time required by using simulator.

These questions were selected to be easy to solve and maximum number of routers does not exceed 3 routers with total of 8 subnets to simplify the answering to the students.

The students spent more time in computations. With increasing of questions difficulty, whereas only few extra seconds are added for the simulator to get the same results.

The results do not put in consideration the hand written mistakes that could occur in solutions with hand while using simulator no error occurs in those results.

The students were asked to put only network ID to reduce solving time and no more extra time spent on other important information or the overall sketch.

While the simulator do all the jobs:

- It gives a detailed table contains, subnet ID ,start IP address ,end IP address ,no. of



host per subnet, ...etc. See Figs. 7, (8.a) and (8.b).

- It gives full detailed answers with a sub netting table (Fig. 7) and topology sketch of all required subnets and WANs connections as shown in Figs. (8.a) and (8.b).

Conclusions and Features of the Simulator

Many conclusions and features in this software can be noticed in this work, as follows:

- 1- The software determines the number of lost IP addresses.
- 2- The simulator validates the entries and prevents the use of incorrect numbers in IP fields (the correct range from 0 to 255).
- 3- By changing the IP address, the software automatically changes the range of network prefix according to IP class (the range of network prefix for class A is from 8 to 30, for class B is from 16 to 30, and for class D is from 24 to 30). Allowing the decision to the administrator to choose the correct subnet mask and to prevent him from putting incorrect subnet mask.
- 4- After entering correct IP address and subnet mask, the software automatically calculates the Network IP address and Broadcast IP address, which are needed by the network administrator in his work.
- 5- The software draws the suggested topology with minimum number of routers.
- 6- The software prevents adding more subnets incase of exceeding the allowable range of IP addresses.
- 7- The software prevents adding more hosts in each subnet incase of exceeding the allowable range of IP addresses.

- 8- After entering the number of hosts in each subnet, the software will descend the order of these subnets automatically. Ordering subnet's hosts from large to small process is required by the VLSM technique.
- 9- The software creates the topology table which contains the important information of addressing required by the network administrator.
- 10- The software makes in consideration the waste IPs for WAN links between routers. Therefore; the administrators does not need to calculate the lost IPs. Moreover the software addresses the links completely and adds it to the topology table.
- 11- The software draws the suggested network with full detailed information for each router's links and subnets.

References

- 1- Cisco Networking Academy Team, "CCNA Exploration Curriculum-Network Fundamentals", Version 4, Cisco System Inc., 2010.
- 2- John Albritton, "Cisco IOS Essentials", McGraw-Hill, 1999.
- 3- Mark A. Sportack, "IP Addressing Fundamentals", Second Edition, Cisco Press, 2003.
- 4- Todd Lammle, "Cisco Certified Network Associate-Study Guide", Seventh Edition, Sybex Inc., 2011.
- 5- Todd Lammle, Sean Odom, Keven Wallace, "CCNP Routing- Study Guide", Sybex Inc., 2001.
- 6- Wendell Odom, "Cisco CCNA Exam-Certification Guide", Cisco press, 2002.

Table 1: IP address classes [1]

Address Class	1 st Octet Range	1 st Octet bits	Net and Host Part of Address	Default Subnet mask	No. of Nets and Hosts/Net
A	1 - 127*	00000000 – 01111111	N.H.H.H	255.0.0.0	128 nets (2 ⁷) 16,777,214 host (2 ²⁴⁻²)
B	128 - 191	10000000 – 10111111	N.N.H.H	255.255.0.0	16,384 nets (2 ¹⁴) 65,534 hosts (2 ¹⁶⁻²)
C	192 - 223	11000000 – 11011111	N.N.N.H	255.255.255.0	2,097,150 nets (2 ²¹) 254 host (2 ⁸⁻²)
D	224 - 239	11100000 – 11101111	NA (multicast)		
E	240 - 255	11110000- 11111111	NA (experimental)		

*All zeros (0) and all ones (1) are invalid host addresses

Table 2: Default subnet masks

Address Class	Dotted-Decimal Form	Dotted-Binary Form
Class A	255.0.0.0	11111111.00000000.00000000.00000000
Class B	255.255.0.0	11111111.11111111.00000000.00000000
Class C	255.255.255.0	11111111.11111111.11111111.00000000

Table 3: Subnet masks

Borrowed Bits	Dotted-Decimal Form	Dotted-Binary Form
1	255.255.255.128	11111111.11111111.11111111.10000000
2	255.255.255.192	11111111.11111111.11111111.11000000
3	255.255.255.224	11111111.11111111.11111111.11100000
4	255.255.255.240	11111111.11111111.11111111.11110000
5	255.255.255.248	11111111.11111111.11111111.11111000
6	255.255.255.252	11111111.11111111.11111111.11111100

Table 4: Class B network subnetting

Network Mask	Prefix Length	Subnet Bits	Node Bits	Subnets	Hosts
255.255.0.0	/16	0	16	0 (1 Net)	65534
255.255.128.0	/17	1	15	2	32766
255.255.192.0	/18	2	14	4	16382
255.255.224.0	/19	3	13	8	8190
255.255.240.0	/20	4	12	16	4094
255.255.248.0	/21	5	11	32	2046
255.255.252.0	/22	6	10	64	1022
255.255.254.0	/23	7	9	128	510
255.255.255.0	/24	8	8	256	254
255.255.255.128	/25	9	7	512	126
255.255.255.192	/26	10	6	1024	62
255.255.255.224	/27	11	5	2048	30
255.255.255.240	/28	12	4	4096	14
255.255.255.248	/29	13	3	8192	6
255.255.255.252	/30	14	2	16384	2

Table5: Class B network subnetting

Network Mask	Prefix Length	Subnet Bits	Node Bits	Subnets	Hosts
255.255.255.0	/24	0	8	0 (1 Net)	254
255.255.255.128	/25	1	7	2	126
255.255.255.192	/26	2	6	4	62
255.255.255.224	/27	3	5	8	30
255.255.255.240	/28	4	4	16	14
255.255.255.248	/29	5	3	32	6
255.255.255.252	/30	6	2	64	2

Table 6: VLSM example

Decimal	255	255	255	252
Binary	11111111	11111111	11111111	11111100

Table 7: Block sizes

Prefix	Mask	Host	Block Size
/25	128	126	128
/26	192	62	64
/27	224	30	32
/28	240	14	16
/29	248	6	8
/30	252	2	4

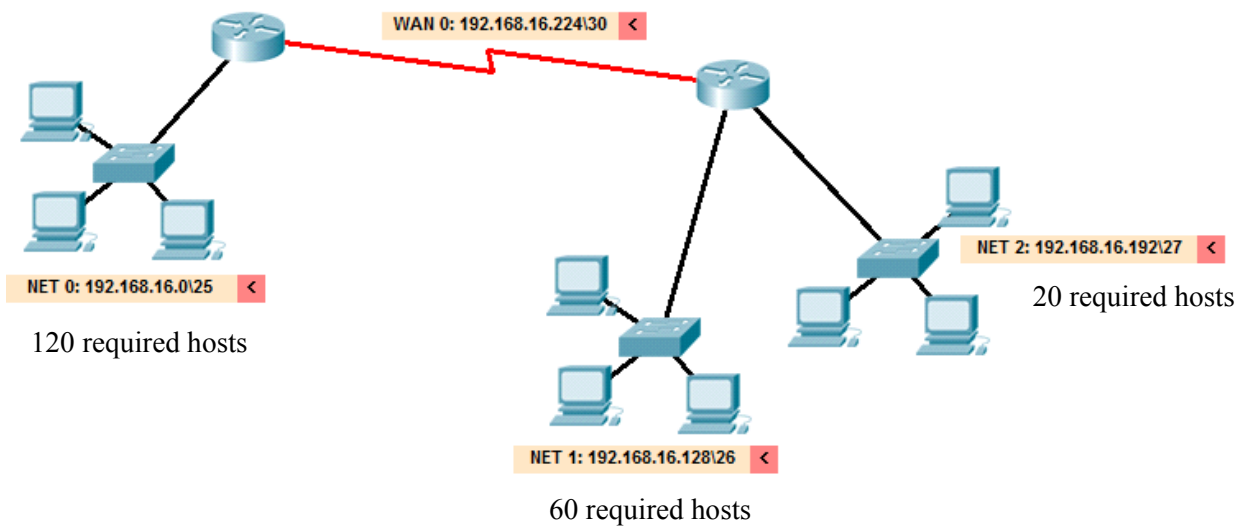


Fig. 4: Desired subnetted network

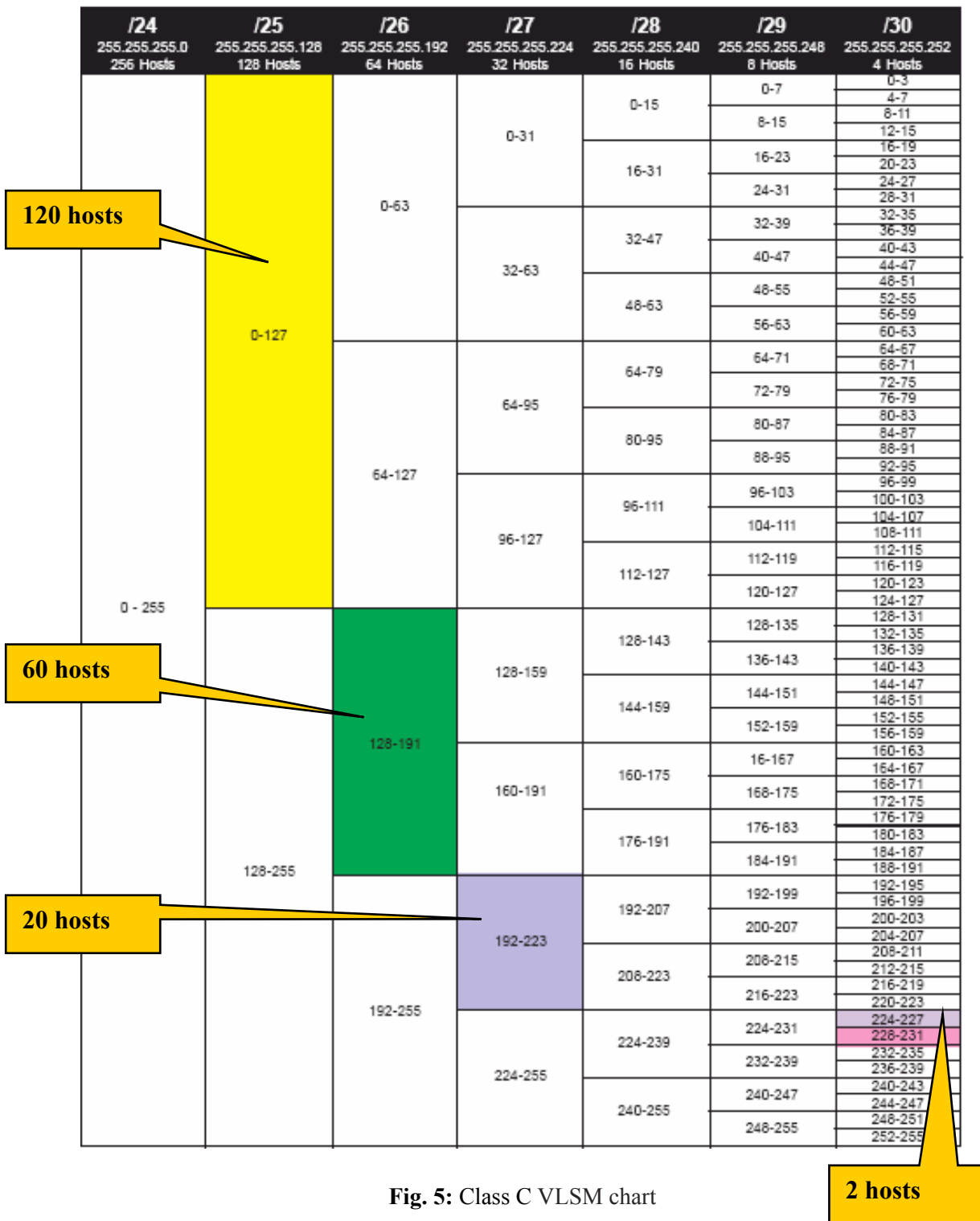


Fig. 5: Class C VLSM chart

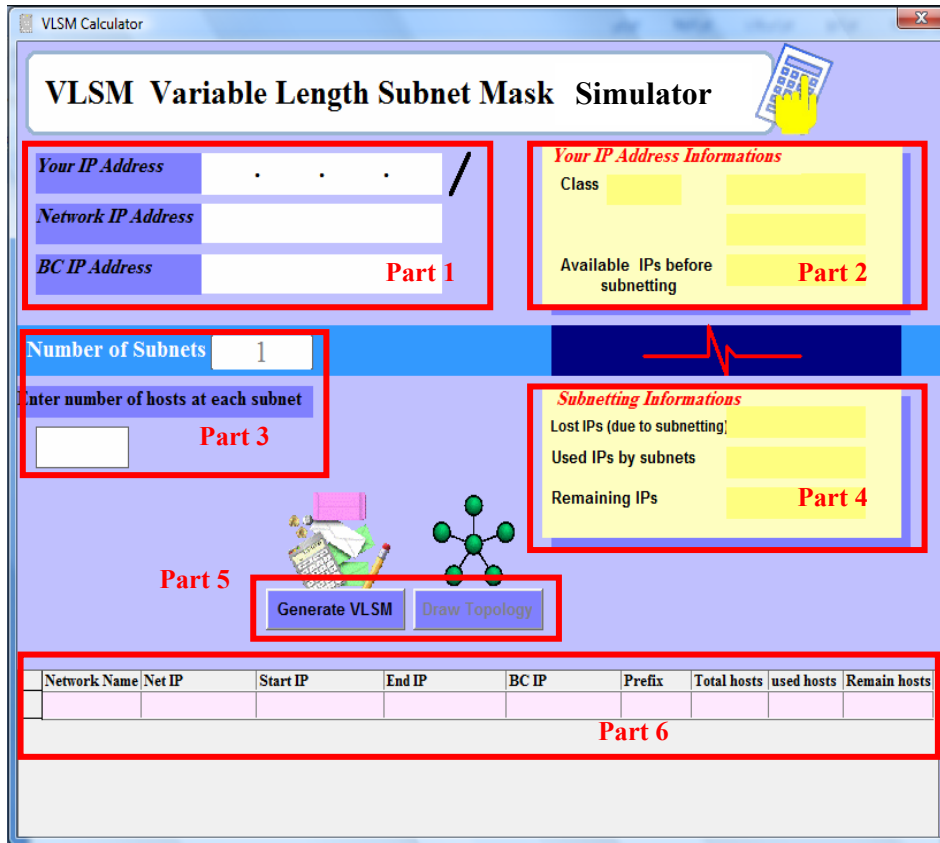


Fig. 6: The main window of the VLSM simulator

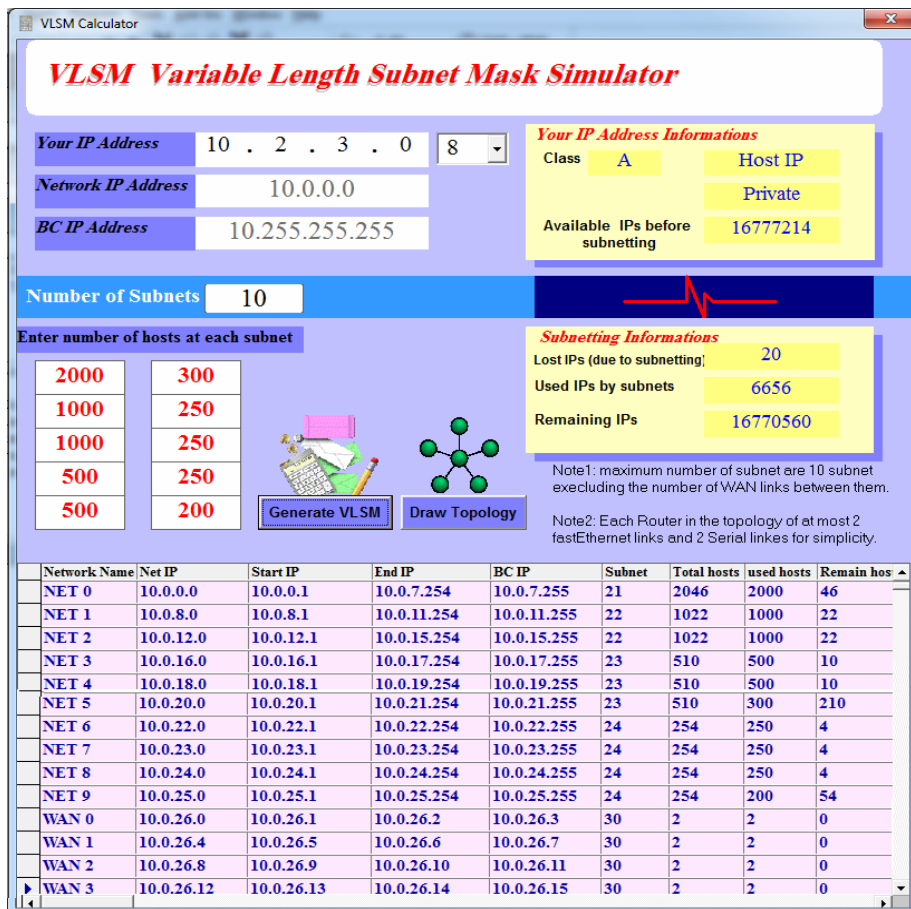


Fig. 7: Entering the IP address and generate VLSM

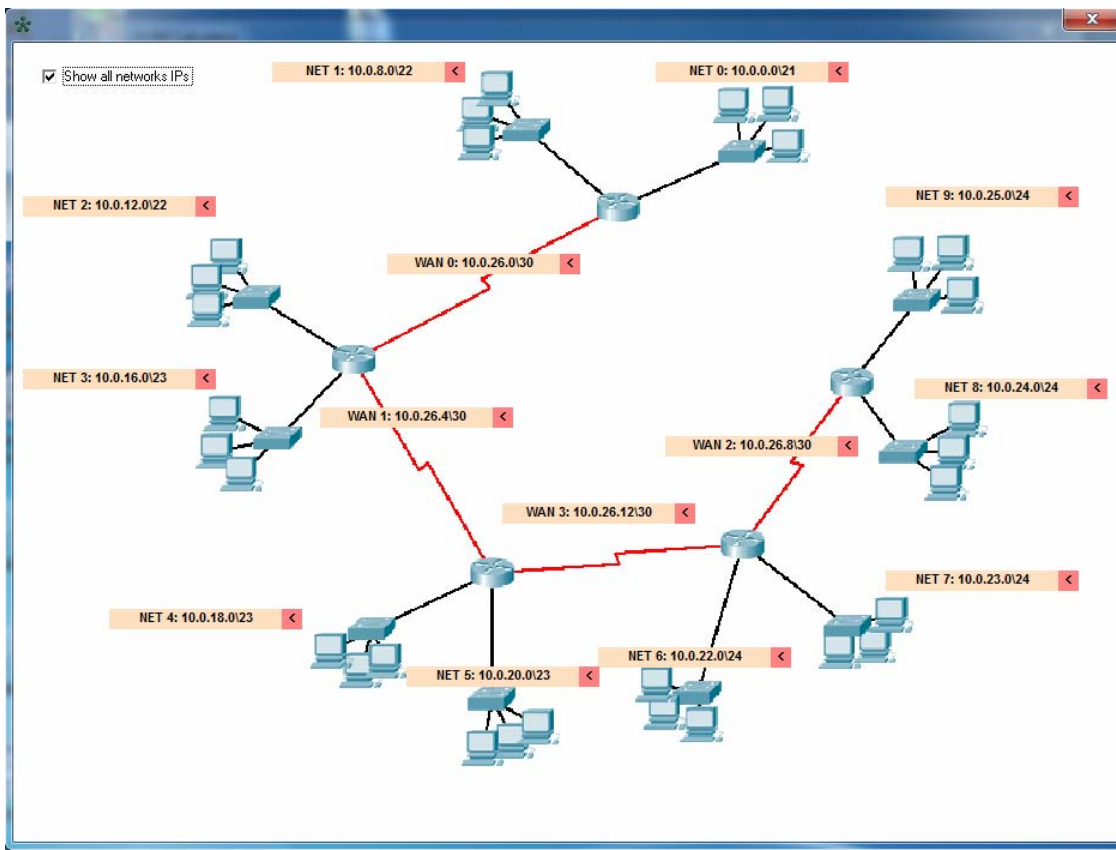


Fig. 8.a: "Draw topology" window with (10 subnets, 4 WAN links)

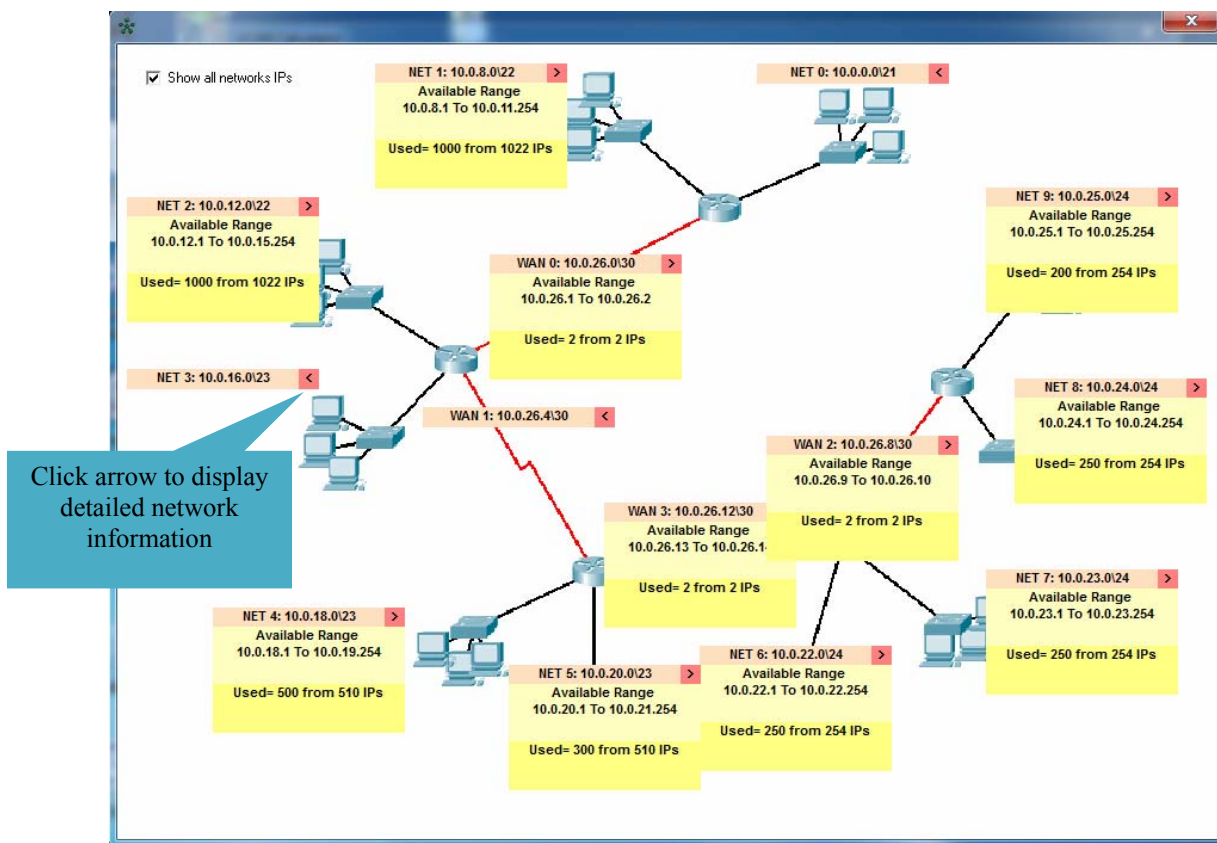


Fig. 8.b: "Draw topology" window with (10 subnets, 4 WAN links)
Shows details by clicking the arrow near subnet's name



Table 8: Students time results for each question (in minutes)

Student No.	Question1		Question2		Question3		Question4	
	hand	simulator	hand	simulator	hand	simulator	hand	simulator
1	12	2	5	1	7	0.5	7	0.5
2	9	1	5	1	3	1	4	1
3	15	1	9	0.5	5	1	6	1
4	13	1	5	1	3	1	3	1
5	12	0.5	13	2	17	1	8	2
6	13	1	9	2	8	2	5	2
7	17	1	8	1	6	1	11	1
Average	13.00	1.07	7.71	1.21	7.00	1.07	6.29	1.21

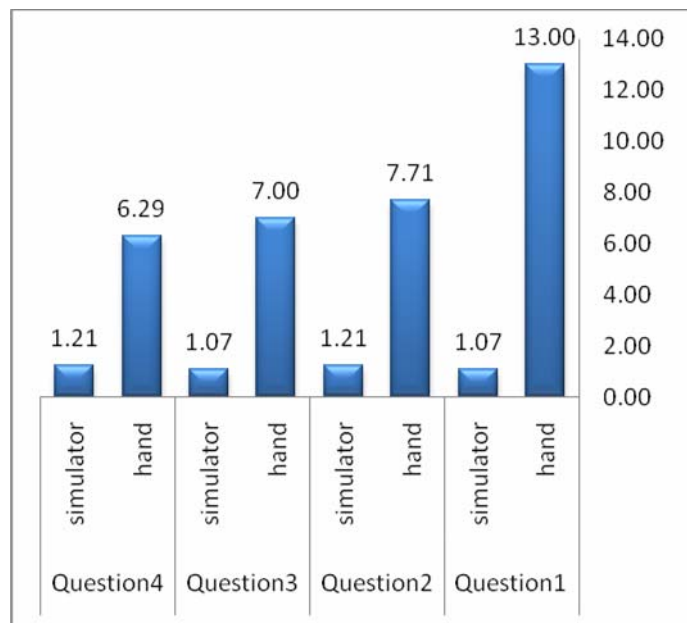


Fig. 9: Chart shows comparison between average student's times for all question result's (in minutes)

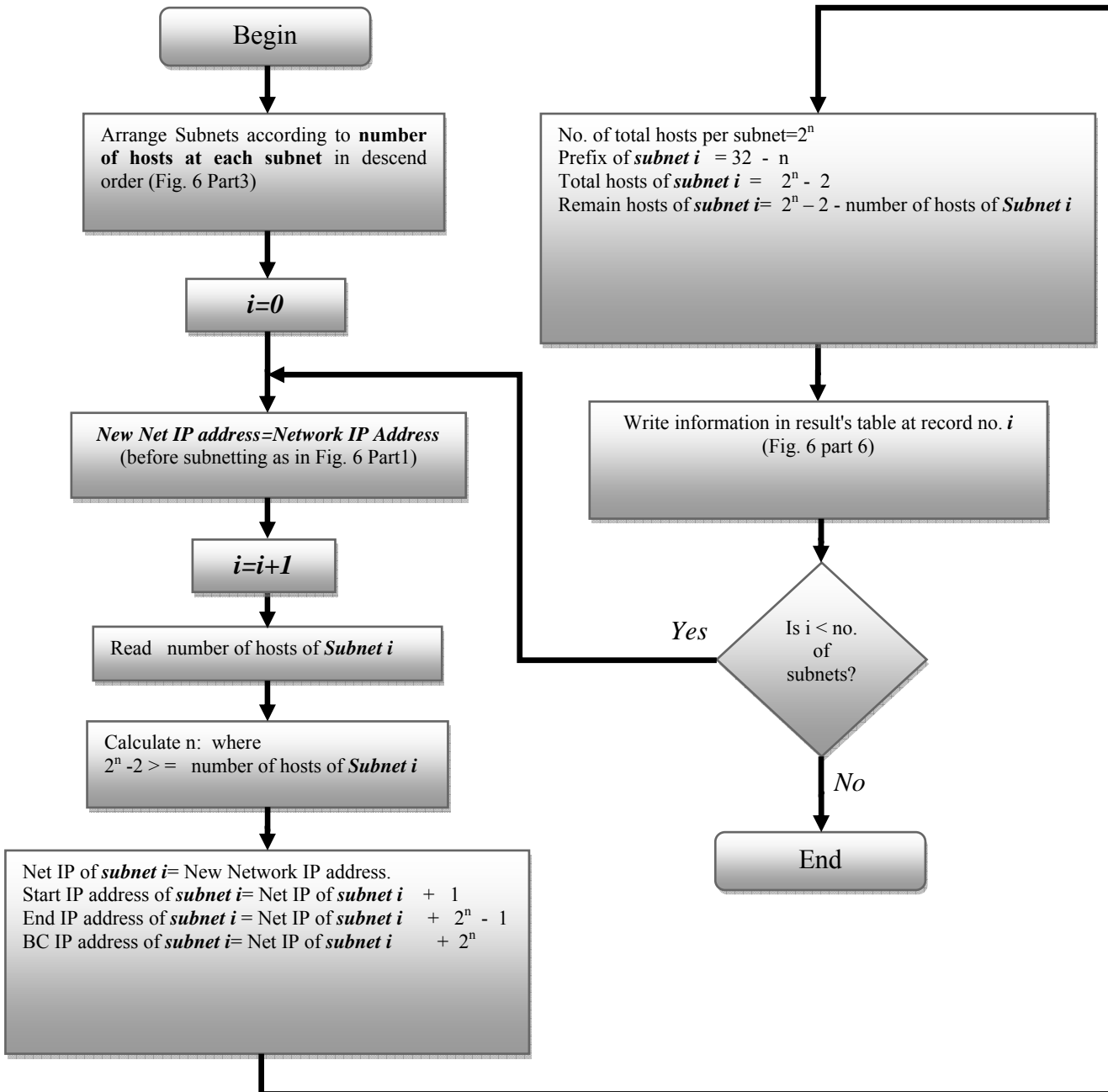


Fig. 10 The flow chart of the generate VLSM process