Ивелина Балабанова

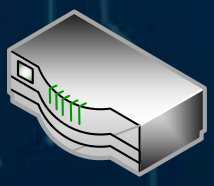
Георги Георгиев

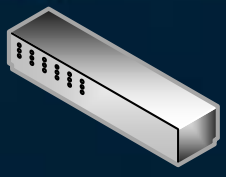
Cisco

МОДЕЛИРАНЕ НА ТРАФИКА В КОМУНИКАЦИОННИ И КОМПЮТЪРНИ МРЕЖИ

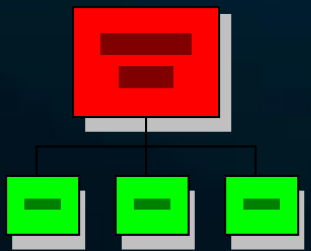
Методично ръководство за лабораторни упражнения

Packet





Tracer



УНИВЕРСИТЕТСКО ИЗДАТЕЛСТВО
ГАБРОВО
2024

ИВЕЛИНА БАЛАБАНОВА **ГЕОРГИ ГЕОРГИЕВ**

МОДЕЛИРАНЕ НА ТРАФИКА В КОМУНИКАЦИОННИ И КОМПЮТЪРНИ МРЕЖИ

МЕТОДИЧНО РЪКОВОДСТВО ЗА ЛАБОРАТОРНИ УПРАЖНЕНИЯ

УНИВЕРСИТЕТСКО ИЗДАТЕЛСТВО
"ВАСИЛ АПРИЛОВ"
ГАБРОВО, 2024

Моделиране на трафика в комуникационни и компютърни мрежи

Методично ръководство за лабораторни упражнения

Рецензент: доц. д-р инж. Боян Димитров Карапенев

За контакти:
Габрово, 5300, ул. „Хаджи Димитър“ №4
Технически университет – Габрово
Катедра „Комуникационна техника и технологии“

ISBN: 978-954-683-715-8

ПРЕДГОВОР

Коректното логическо и физическо планиране, изграждане, експлоатиране и поддържане на мрежови трасета спрямо спецификата и изискванията на средата е основна задача в съвременните комуникации. Проектирането и използването на подходящи приемо-предавателни комуникационни звена, софтуерни инструменти и приложения за мониторинг на входящия и изходящия информационен пренос е друг аспект, свързан с цялостното поддържане на оптимални нива на обслужвания трафик. Същевременно с това от друга страна е жизнено важно да бъде гарантирана сигурността на данните чрез прилагане на адекватни подходи за администриране и управление на организацията на достъпа при съответни потребителски роли до информационни системни ресурси на персонално, по отдели и цялостно корпоративно ниво.

Това стои в основата за поддържането на цялостта и високото бързодействие, както и значителното минимизиране на факторите за забавяне и вероятността за загуба при доставка на пакети между крайните потребители. Поради това е необходимо много добро познаване на функционалността и особеностите на различните видовете мрежови устройства, технологии, стандарти, типовете комуникационни интерфейси, основните аспекти на маршрутизацията, алгоритмите за криптиране и подходите за обезпечаване на сигурността на информацията.

Всички посочени дейности могат да се дефинират като основна предпоставка за подготовка на висококвалифицирани специалисти, извършващи дейности на различни нива в сферата на управлението и обслужването на информационно-комуникационните системи с разнородно предназначение в бизнеса и индустрията. В тази връзка в настоящото ръководство по дисциплините „Телекомуникационни мрежи и протоколи“ и „Комуникационни протоколи“ са разработени лабораторни упражнения в следните направления:

- *информационно-комуниникационни звена и канали за връзка при различни типове трафик на основата на статистически подходи с изследване на основни параметри и типове обработки върху предаваните сигнали на каналите;*
- *жични и безжични мрежови конфигурации с демонстриране на основни аспекти на статичната и динамичната маршрутизация при конкретни стандарти и протоли;*
- *оптични комуникации и процедури по оптимизиране на параметри на оптични кабелни трасета;*
- *обезпечаване на сигурността на основата на виртуални локални и частни мрежи, криптиране на трафика, управление на достъпа WEB и Email съдържание;*
- *Управление на предаването на гласови данни между IP абонсти посредством глобалната Интернет мрежа;*
- *IoT и облачни технологии в концепциите "Интелиген дом", "Сензорни системи за сграда автоматизация", "Системи за пожароизвестяване и пожарогасене", "Администриране на Email трафика между корпоративни бизнес клиенти" и др.*

Авторите изказват най-сърдечни благодарности към Рецензента доц. д-р инж. Боян Димитров Карапенев по повод направените препоръки при крайното оформяне на ръководството!

Лабораторно упражнение № 1

ИЗСЛЕДВАНЕ И СИМУЛАЦИОННО МОДЕЛИРАНЕ НА ИНФОРМАЦИОННО-КОМУНИКАЦИОННИ СИСТЕМНИ С ОПАШКОВА ОРГАНИЗАЦИЯ НА ОБСЛУЖВАНЕТО В СРЕДА JMVA

1.1. Цел на упражнението

Да се запознаят студентите е прилагани аналитични подходи и алгоритми за имитационно моделиране на телекомуникационни и информационни инфраструктури, както и оптимизиране на основни индекси на производителност за подобряване на услугата QoS.

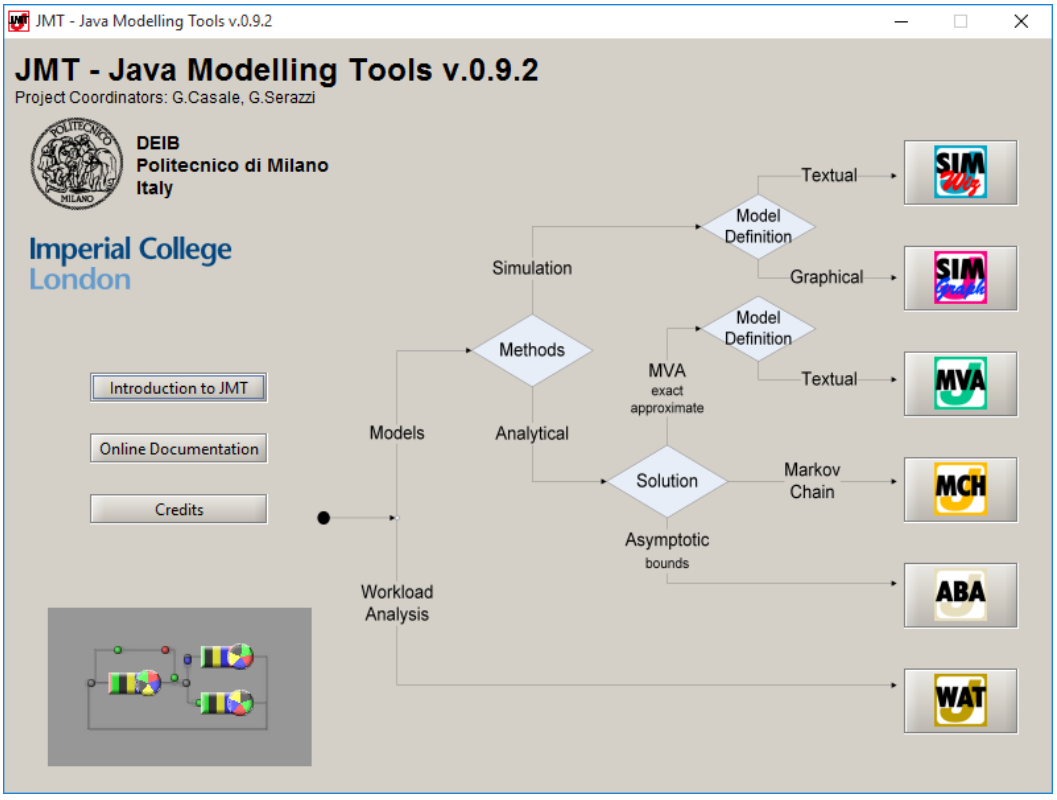
1.2. Теоретична постановка

Симулационна среда JMVA за моделиране на мрежи с опашки

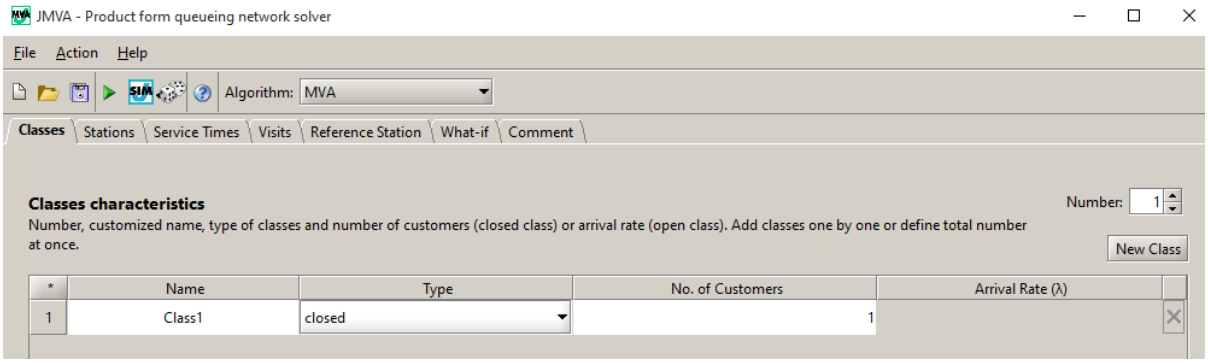
JMVA е част от групата симулатори, включени в среда JMT (Java Modeling Tool), даващ възможности на потребителите за прилагане на технически подходи за системно моделиране и параметрично апроксимиране на ИКТ структури с опашкова организация на заявките. Интерфейси при зареждане на JMT и JMVA са показани на фиг. 1.1. Средата позволява създаване и конфигурация на "*open*", "*closed*" и "*mixed*" мрежи с опашки на основата на "Exact" и „Approximate" категории от алгоритми.

В групата на *моделиращите технически подходи* за телекомуникационни инфраструктури от тип "closed" спадат, както следва:

- MVA (Mean-Value Analysis) [RL80],
- RECAL (Recursion by Chain Algorithm) [CG86];
- MoM (Method of Moments) [Cas06] [Cas11];
- CoMoM (Class-Oriented Method of Moments) [Cas09];
- RGF (Recursive Generating Function) [HL04].



а)



б)

Фиг. 1.1. *Интерфейси на симулационната среда JMT и JMVA за моделиране и параметрично апроксимиране на мрежови структури с опашкова организация*

Когато създаваните модели съдържат "*Load Depended Station*" или „*станция, при която следва фиксиране на обслужващото време и броя на постъпванията на потребителски заявки*", то MVA алгоритъмът се разглежда като [BBC+81] вариант на класическата MVA версия [1].

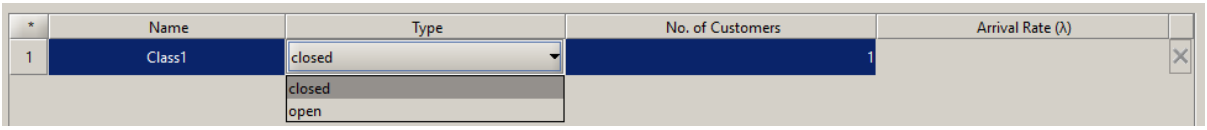
Във връзка с поддържаните *апроксимиращи технически подходи* или AMVA алгоритми са заложени:

- ✓ Chow [Cho83];
- ✓ Bard-Schweitzer [Sch79];
- ✓ Linearizer [CN82];
- ✓ DeSouza-Muntz Linearizer [dSeSM90];
- ✓ Aggregate Queue Length (AQL) [ZES88];
- ✓ Monte Carlo [KW97].

Важна особеност е прилагането на подходите за параметрично апроксимиране при опашкови мрежови структури от тип „closed". По дефиниция тези подходи главно са използвани за оценка на „дължината на опашката" относно структурните станции, обслужващи постъпилите потребителски заявки [2].

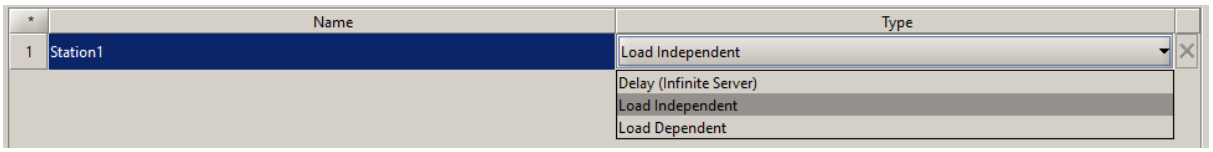
Дефинирането на модел преминава през последователна настройка на следните параметри:

- *Classes* – задаване на типа на ИКТ опашковата структура (фиксиране на „*количество потребители* ***N*** *(брой системни обслужващи станции)*" за „*closed*" и „*скорост или интензивност на постъпване на потребителски заявки* ***λ***" за „*open*" тип телекомуникационни системи) – фиг. 1.2;



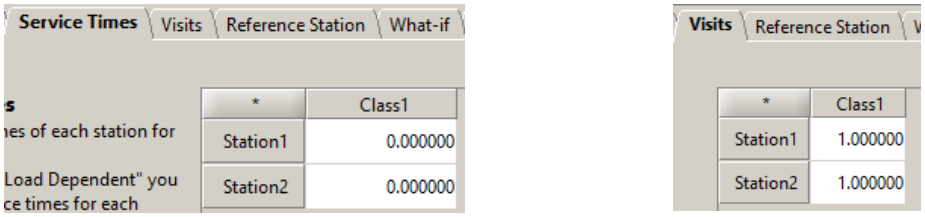
Фиг. 1.2. *Дефиниране на потребителски системен клас*

- *Stations* – указване на обслужващите центрове относно системните потребителските заявки – фиг. 1.3;



Фиг. 1.3. *Назначаване на обслужващи системни станции*

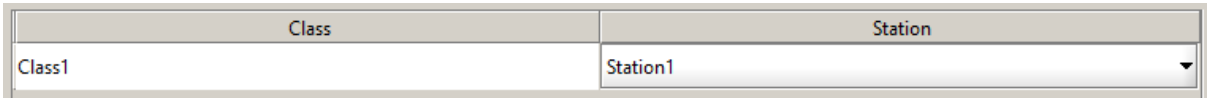
- *Service times and Visits* – дефиниране на времената на обслужване и на броя на постъпванията за всяка структурна станция – фиг. 1.4;



а) *б)*

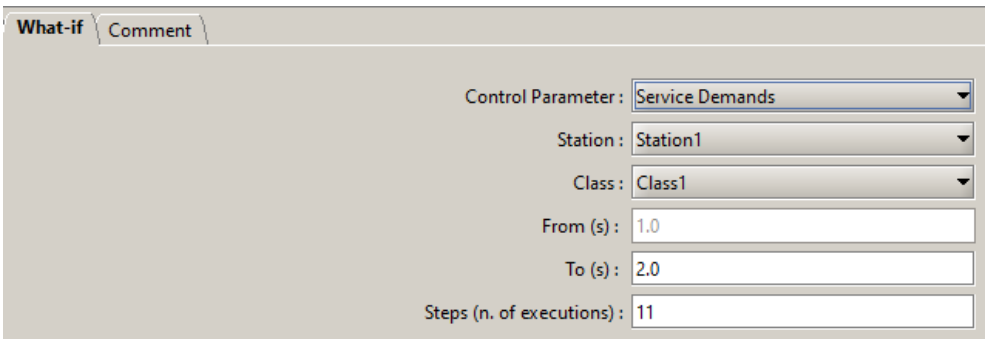
Фиг. 1.4. *настройка на Service demand – а) Service time и б) Visits)*

- *Reference Stations* – представляват “User station” или още “Delay station” („потребителка“ или още „закъснителна“ станция). Разглеждат се като външни звена източници на потребителските заявки, имащи особено отношение към адекватния и коректния избор на конфигурационни параметри относно „Пропускателната способност“ и „Сумарното време за престой в опашката и обслужване“ на системно ниво – фиг. 1.5;



Фиг. 1.4. *Настройка на Reference stations*

- *What-if Analysis* – дава възможност за изпълнение на серия от симулации на модела и изследване на основни индекси на производителност на тестовите модели съобразно различни контроли параметри (фиг. 1.5):
  - Number of Customers;
  - Arrival Rates;
  - Population Mix – дефинира се като съотношение между броя на потребителите за избран “class i” спрямо общото количество системни потребители;
  - Service Demand (Service times and Visits).



Фиг. 1.5. *Конфигуриране на What-if процедура*

Фиг. 1.6. *Въвеждане на описание*

- *Comment* – вмъкване на пояснящ и описателен коментар относно спецификата и особеностите на създавания мрежови модел с опашкова организация – фиг. 1.6.

JMVA позволява изследване на набор от системни параметри, между които са:

Индекси за отделен "class" и "station:

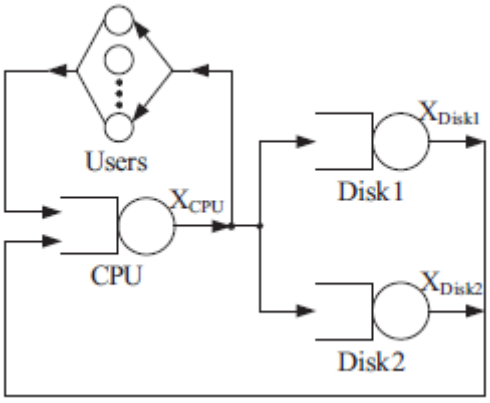
- Throughput – пропускателна способност;
- Query Length – дължина на опашката;
- Residence Time – време за престой и време за обслужване за всиеки постъпвания на потребителски заявки;
- Utilization – количествена процентна част за заетост на системна станция, свързана с изпълнение на потребителски заявки;
- System Power (Throughput / Response time) – отношение на „пропускателната способност" спрямо „време за престой и време за обслужване в рамките на единично постъпване на заявки".

„Сумарни" или „Aggregate" индекси за цялостното поведение на модела:

- System Throughput;
- System Response Time;
- Average number of customers in the systems;
- System Power (Aggregate System Throughput / Aggregate system Response Time).

Мрежов модел с опашкова организация с включване на единичен системен клас от тип "closed"

Фигура 1.7 представя сценарий на мрежова топология на ИКТ модел с опашка с присвояване на единичен клас от тип "*closed*". Моделът съдържа едно закъснително звено или източник на потребителски заявки *Users*, постъпващи в станция за първична изчислителна обработка *CPU*. Резултатите от обслужване на потребителските заявки следва да бъдат повторно обработени и временно съхранени в станции *Disk1* и *Disk2*, след което изпратени към блока за първична обработка, като едновременно с това и изпратени към потребителите, които са ги заявили.



Фиг. 1.7. *Мрежова топология на модел с единичен системен клас от тип "closed"*

За разглеждания ИКТ модел с опашкова организация N = 3, т.е. моделът съдържа три станции с предназначение за обработка на постъпилите заявки. Всяка станция се изгражда от два структурни компонента, респективно:

- Опашка за престой на системните повиквания или потребителските заявки, онагледена на фиг. 1.8.а);
- Блок за обслужване на заявките от опашката, показан на фиг. 1.8.б);

а) *б)*

Фиг. 1.8. *Структурна опашка а) и блок за обслужване на заявки б) при станция CPU*

В таблица 1.1 са дадени конфигурационни параметри при създаване на модела „времената на обслужване“ и „количеството на постъпванията на потребителски заявки“, касаещи всяка структурна станция [1, 2].

Таблица 1.1. Обслужващи времена и постъпвания на заявки относно мрежов модел с опашкова организация с включване на единичен системен клас от тип “closed”

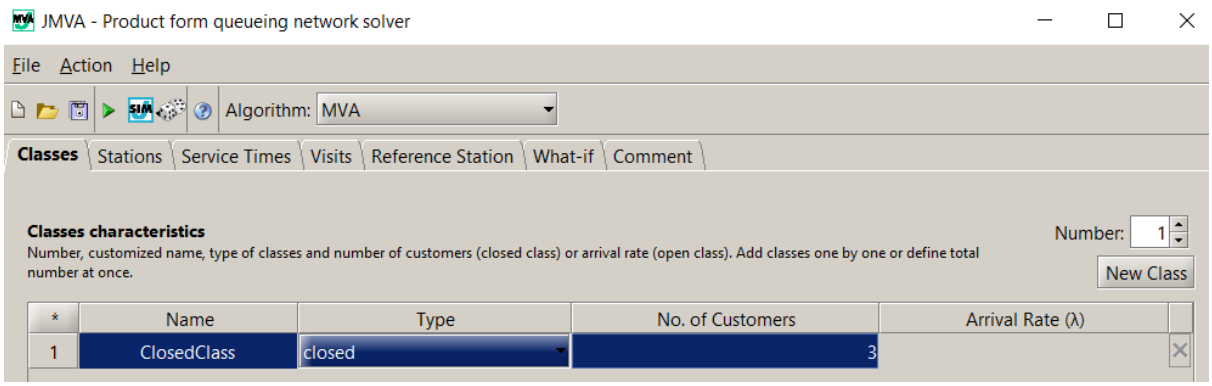
Paremeters	CPU	Disk1	Disk2	Users
Service Times [s]	0.006	0.038	0.030	16.000
Visits	101.000	60.000	40.000	1.000

1.3. Последователност на работа и задачи за изпълнение.

1.3.1. Моделиране и оптимизация на мрежови модел с опашка организация с интеграция на единичен потребителски клас.

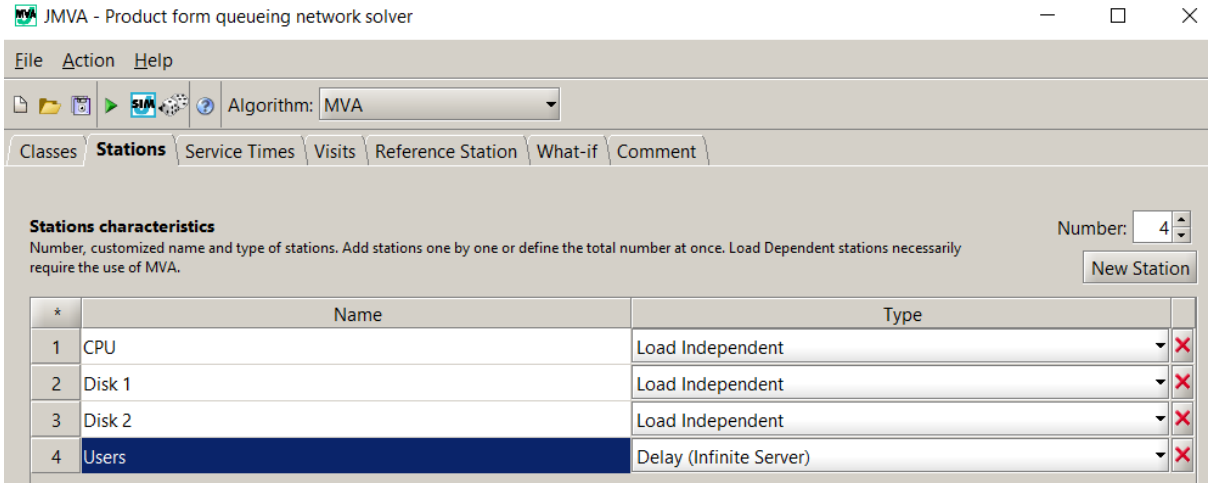
Моделирането и конфигурацията на мрежовия модел с опашкова организация при включване на единичен потребителски клас от тип “closed” със сценарии за извършване на първична и вторична обработка на масиви от данни се състоят в следните стъпки:

- Стартиране на програмната среда JVMA и създаване на нов документ във връзка с настройка на параметрите и избора на технически алгоритми относно модела;
- Дефиниране на потребителски клас от тип “closed” с наименование “ClosedClass” и задаване на броя на потребителите N = 3 относно генерираната интензивност на потока от заявки, показано на фиг. 1.9;



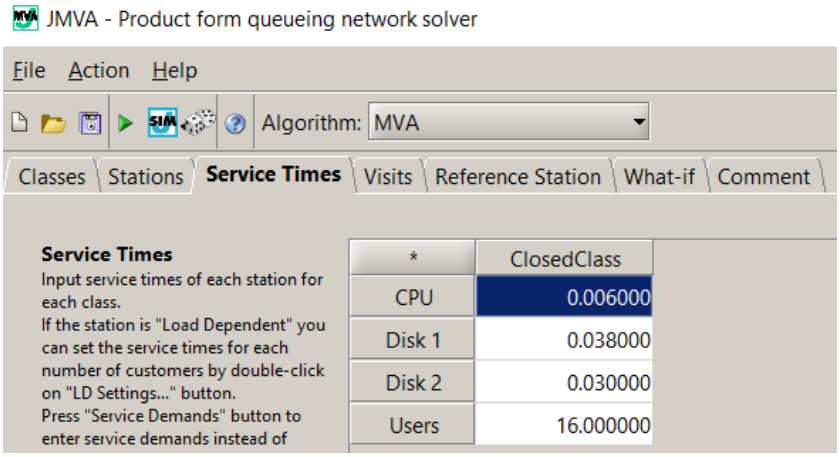
Фиг. 1.9. *Създаване на клас “ClosedClass” и фиксиране на количеството системни потребители N = 3*

- Изграждане на структурата на модела с включване на станции с опашкова организация на постъпващите заявки за последваща процедурна обработка, респективно “CPU”, “Disk 1” и “Disk 2”, и структурна станция източник на информационния трафик “Users”. Оказване на тип „независими от натоварването“ за обслужващите “CPU”, “Disk 1” и “Disk 2” и “Infinite Server” за потребителската закъснителна станция “Users”. Стъпката е дадена на фиг. 1.10;



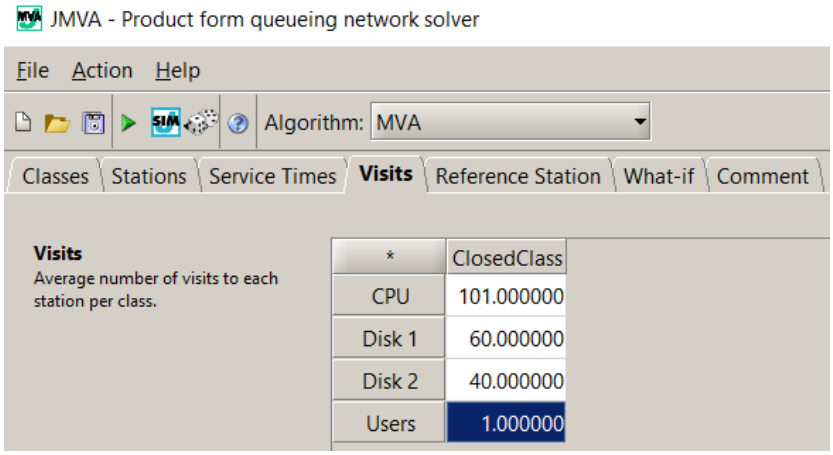
Фиг. 1.10. *Включване на обслужващи “CPU”, “Disk 1” и “Disk 2” и потребителска закъснителна станции (източник на пакетни заявки) “Users”*

- Настройване на времето на обслужване на заявките, ограничени за обработка в структурните опашки, “CPU”, “Disk 1” и “Disk 2” и времевия интервал, в който потребителите е необходимо да изчакат завършване на процедурната обработка на заявките при станция “Users”, представено на фиг. 1.11;



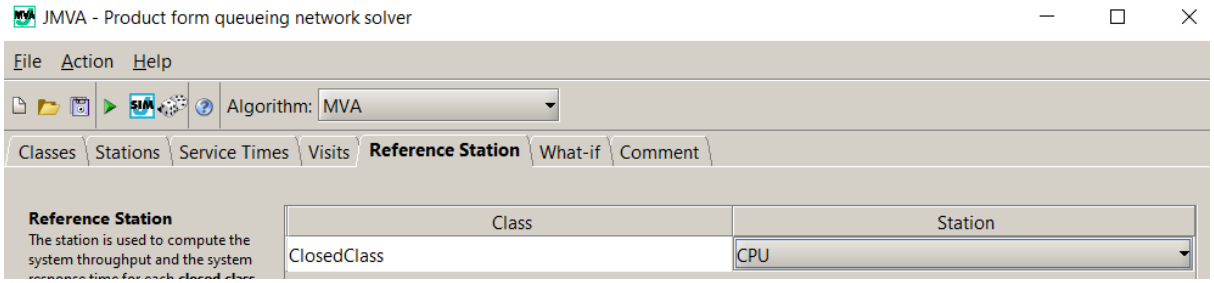
Фиг. 1.11. *Избор на “Service Times” при станции CPU”, “Disk 1”, “Disk 2” и “Users”*

- Прецизиране на количествените постъпвания на потребителските заявки в рамките на обслужващите и закъснителната структурни станции, съответно CPU”, “Disk 1”, “Disk 2” и “Users” по входно-изходните трафични направления, дадено на фиг. 1.12. Трябва да се спомене особеността за общото количество постъпвания или процентното разпределяне на трафика при системните звена за вторична обработка, които сумарно е необходимо да бъдат “100, т.е. 100 %”;



Фиг. 1.12. *Дефиниране на броя на постъпванията на заявки по отношение на станции CPU", "Disk 1", "Disk 2" и "Users"*

- Избор на референтна станция за клас тип "closed", за която да бъдат изчислени телетрафичните „*Системна пропускателна способност*" и „*Системно комплексно време за престой в опашката и обслужване на заявки, ограничени до едно постъпване*", като за конкретния случай тази за първична информационна обработка, е съответно станция "CPU" (фиг. 1.13);

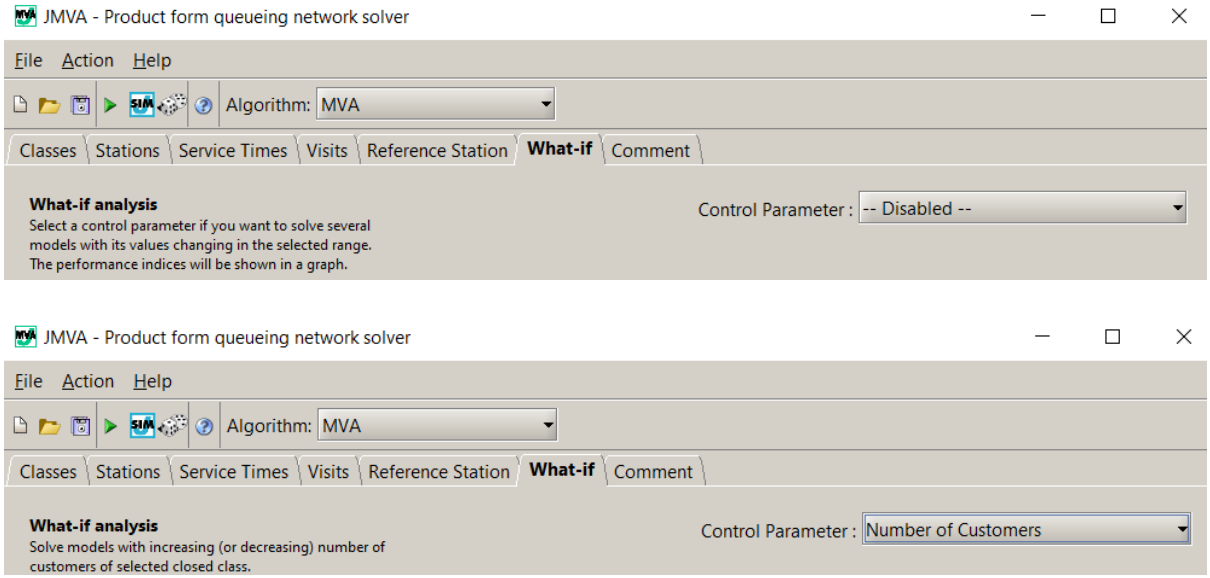


Фиг. 1.13. *Оказване на Reference Station за създаден потребителски клас от тип "Closed" – структурно звено за първична обработка на трафика "CPU, относно калкулация на "System Throughput" и "System Response Time"*

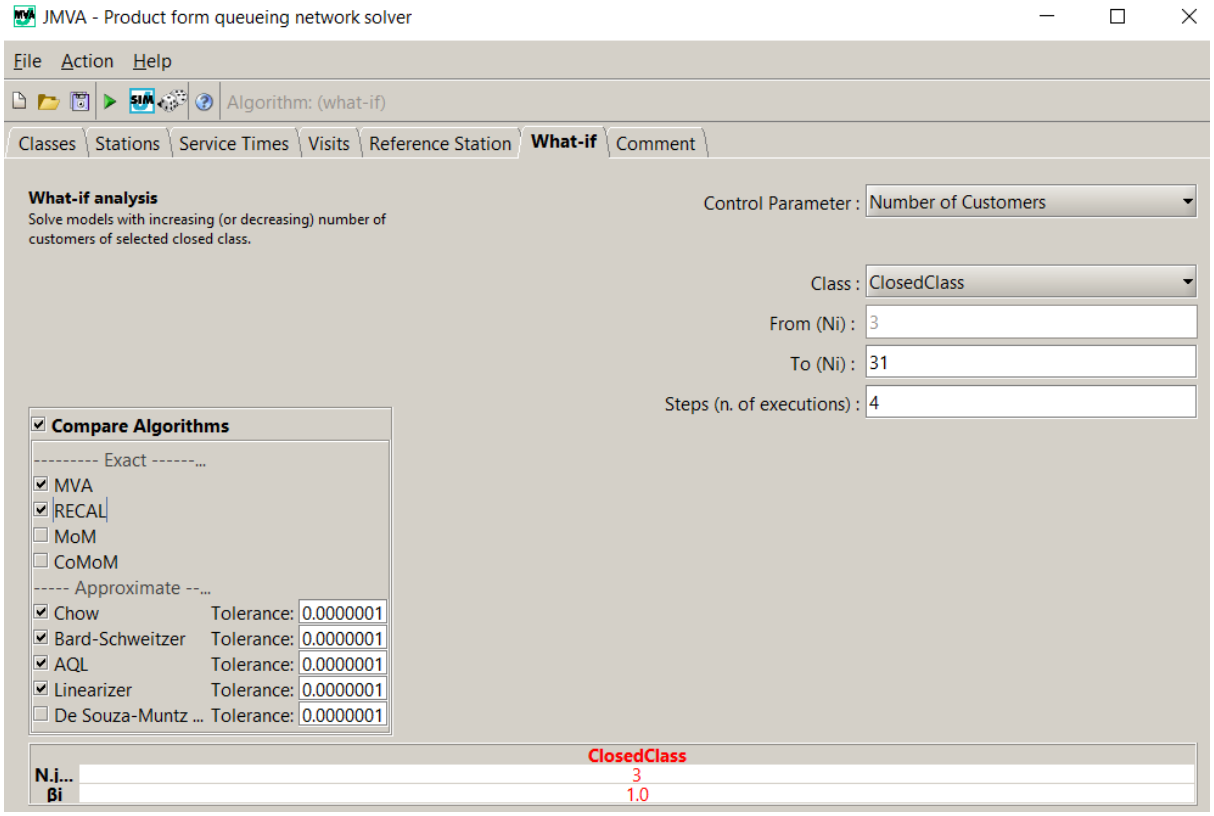
- Активиране на режима "***What-if***", осигуряващ възможност за изпълнение на серия от симулации, в чиято продължителност се изследва цялостното поведение на създадения мрежови модел с опашкова организация. Режимът позволява активация и приложение спрямо група от *контролни параметри* като за конкретния случай може да бъде указан „*Number of Customers*" на фиг. 1.14;

- Настройка на показателите по отношение на указания контролен параметър „*Number of Customers*" – „начален" и „краен брой потребители", както и „стъпка на изпълнение". Разрешаване на опция „**Compare Algorithms**" и избор на технически „**Exact**" – ***MVA, RECAL***, и "**Approximate**" – ***Ghow, Bard-Schweitzer, AQL, Linearizer***, видове алгоритми за моделиране и оптимизация на параметри на трафика на телекомуникационна мрежова структура с опашкова организация на заявките. По отношение на създадения потребителски клас "ClosedClass" индикаторът **βi**, обвързан с

процеса на симулационно изпълнение, се настройва автоматично. Процедурата е онагледена на фиг. 1.15;



Фиг. 1.14. *Избор на контролен параметър за активация на симулационен режим "What-if"*

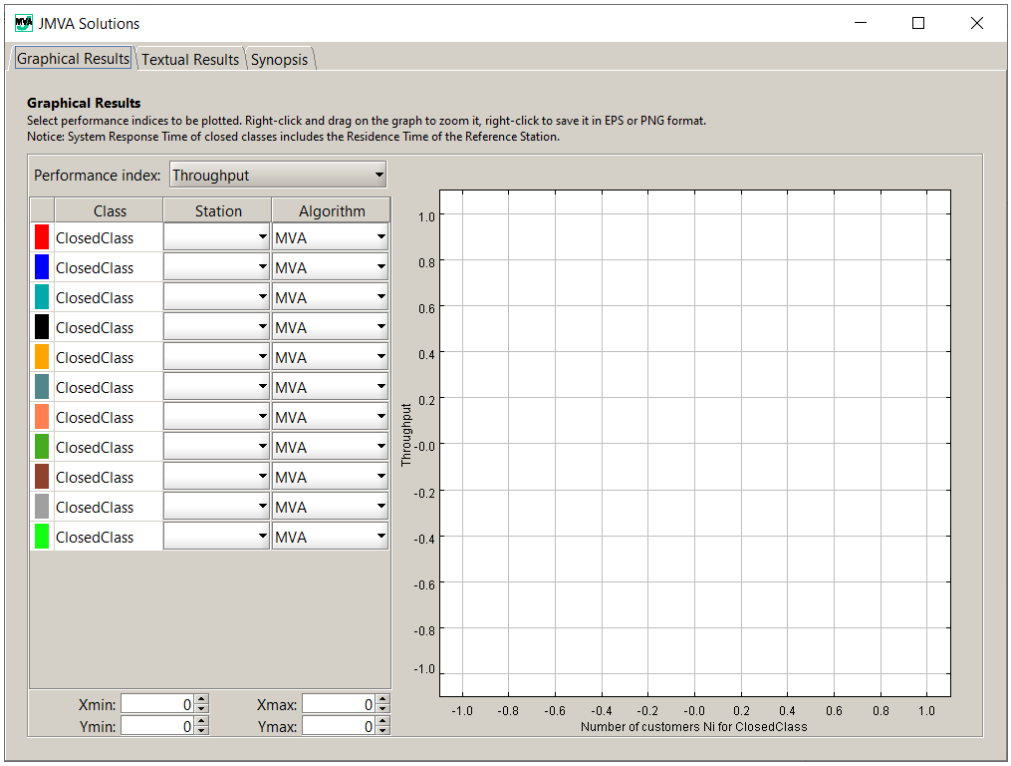


Фиг. 1.15. *Задаване на начален и краен брой потребители, "Exact" и "Approximate" алгоритми относно параметри на трафика при системен "What-if" анализ*

- Стартиране на симулационните процеси и анализ на категории от резултатни осцилограми съобразно приложените "Exact" и "Approximate" технически

подходи за желани трафични параметри на мрежовия модел с опашкова организация. На фиг. 1.16 е показан началният изглед на графичния потребителски интерфейс за избор на:

- *Индекс на производителност;*
- *Потребителски клас;*
- *Структурна станция;*
- *Моделиращ вид алгоритъм.*

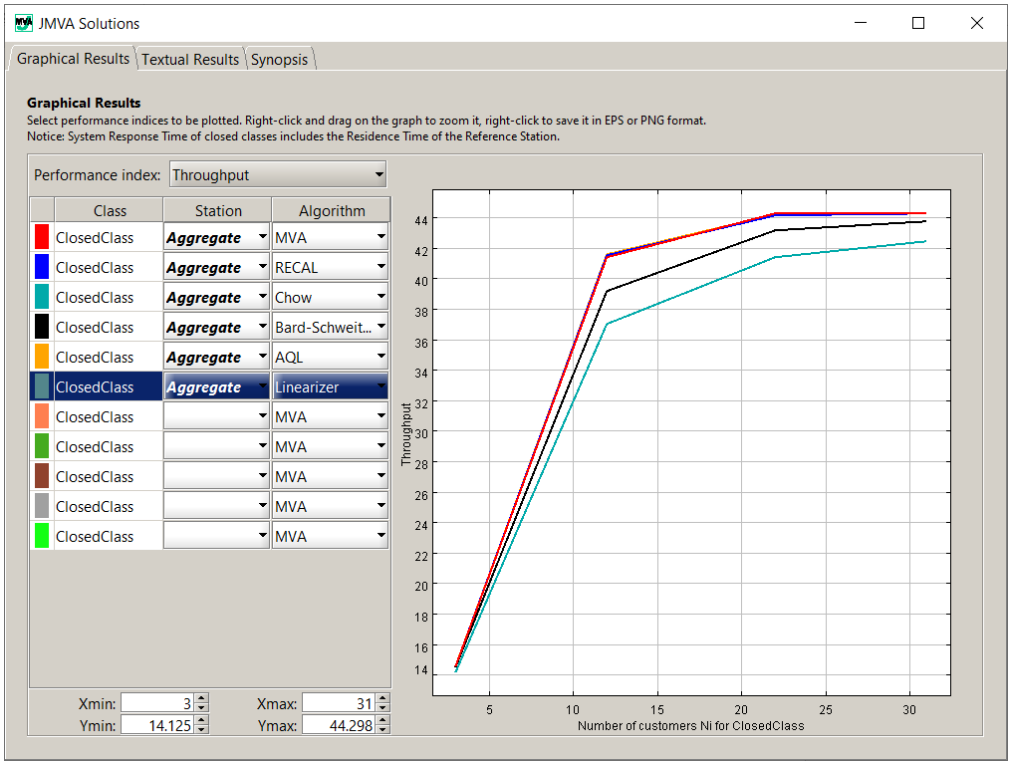


Фиг. 1.16. *Интерфейс на графичния инструмент за извеждане на симулационните резултати съобразно фиксирани контролни параметри и специфицирани структурни звена*

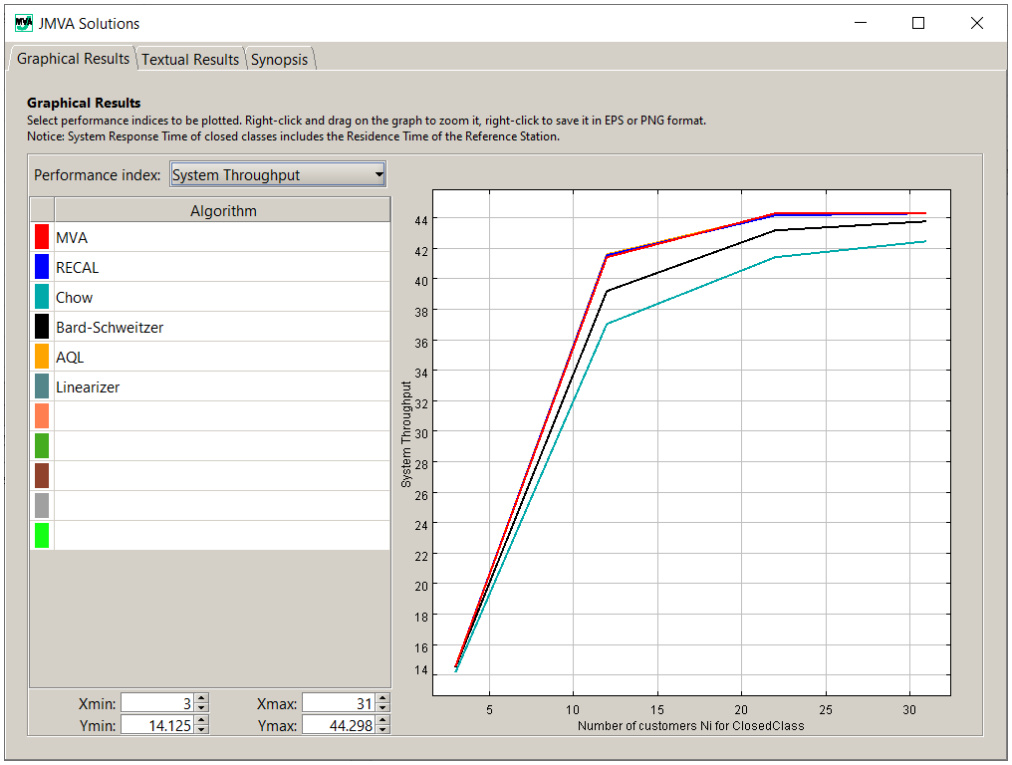
1.3.2. Извеждане, анализ и оценка на резултатите от симулация на мрежови модел с опашка при с интеграция на единичен потребителски клас

Следва да се генерират следните семейства от характеристики на целеви трафични параметри и *установят тенденциите в поведението* на модела и отделно за всяка конкретна обслужваща станция:

- „Системната пропускателна способност“- фиг. 1.17;
- „Системно сумарно време за чакане в опашката и обслужване в рамките на еднократно постъпване“ – фиг. 1.18;
- „Пропускателна способност“ за системни обслужващи звена “CPU’, “Disk 1” и “Disk 2” – фиг. 1.19;
- „Комплексни времена за престой в опашката и обслужване за всички постъпвания на заявки“ за обработващи звена на информационния трафик “CPU’, “Disk 1” и “Disk 2” – фиг. 1.20;
- „Отношение „Пропускателна способност“ и „Сумарно време за престой в опашката и обслужване при единично постъпване““ – фиг. 1.21.

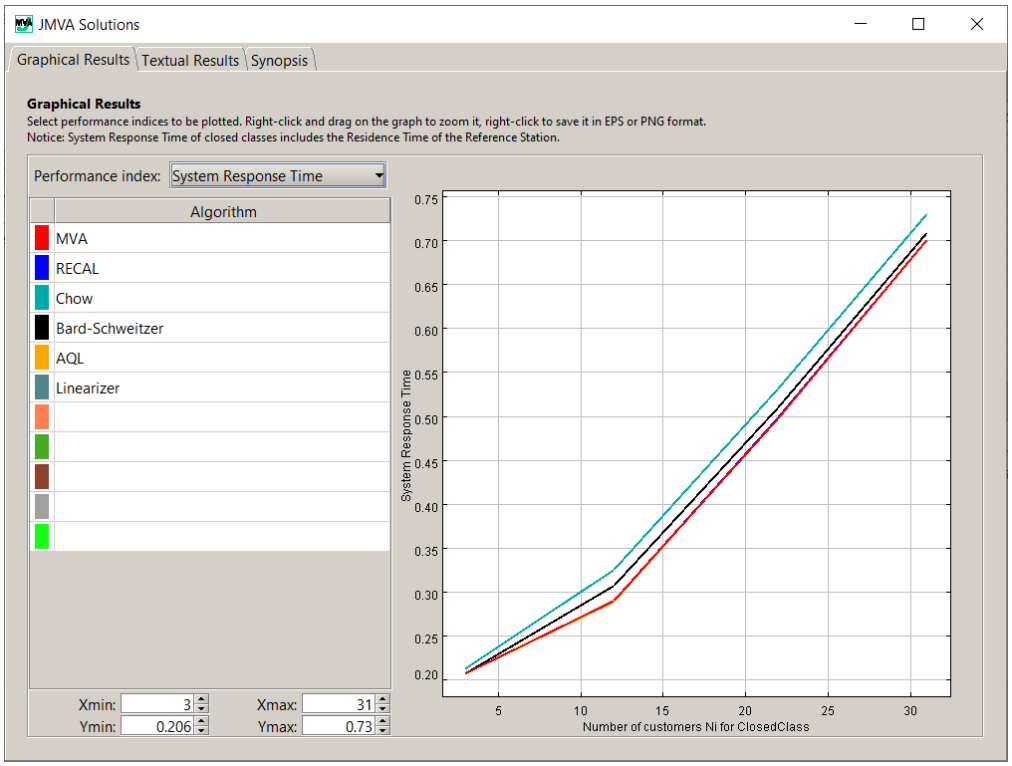


а)

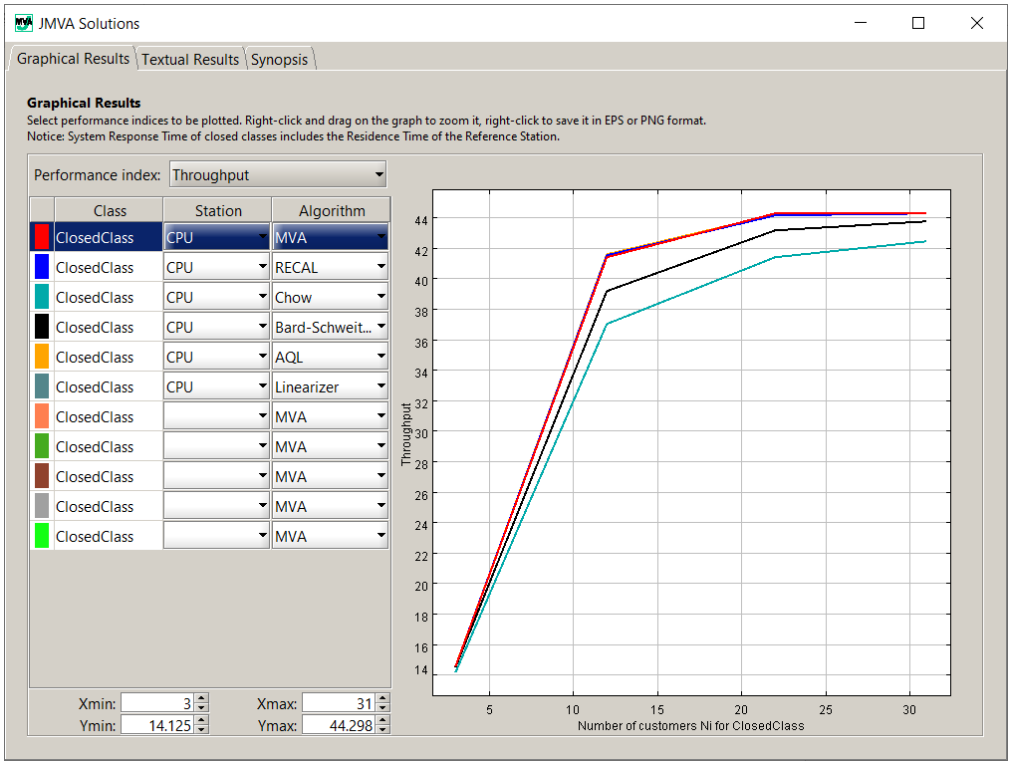


б)

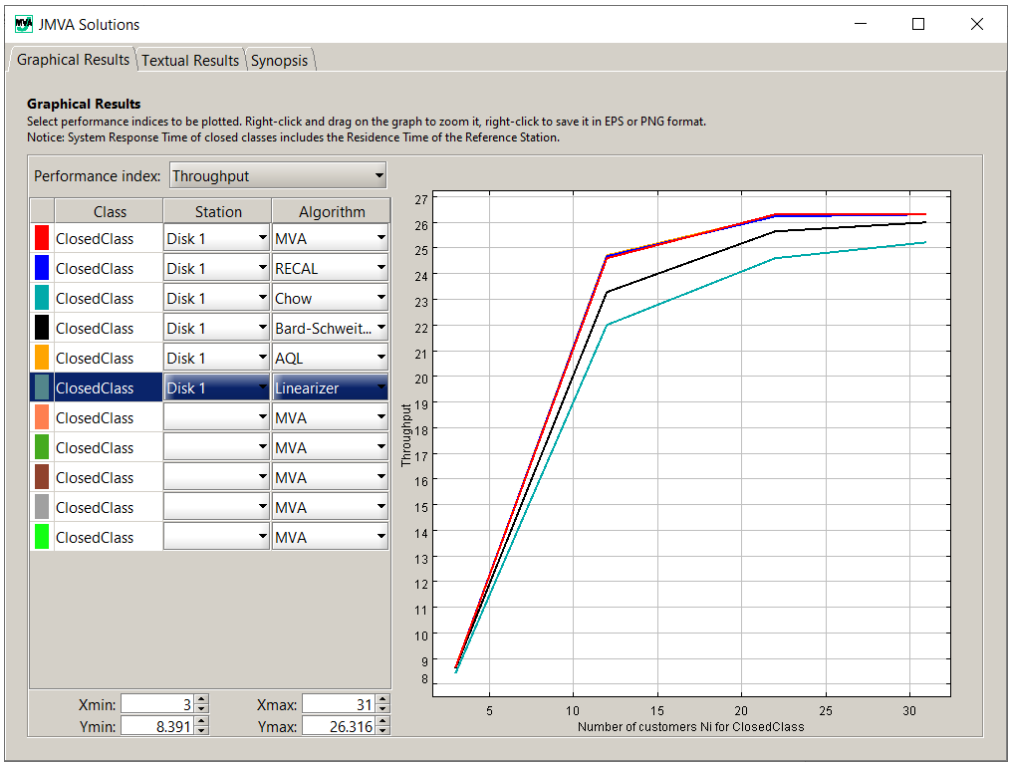
Фиг. 1.17. *„Системна пропускателна способност“ на модела при “Exact” и “Approximate” алгоритъм – при избор на опция „Aggregate“ и „System Throughput“*



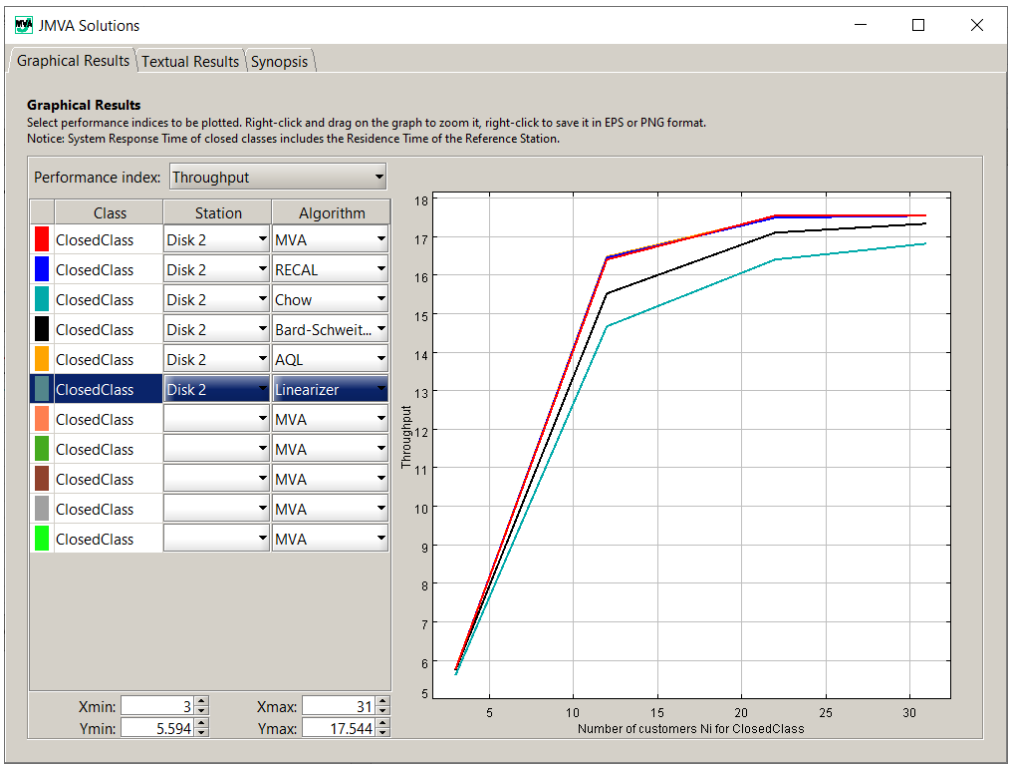
Фиг. 1.18. *„Системно комплексно време за престой в опашката и обслужване в рамките на едно постъпване“ на модела при “Exact” и “Approximate” алгоритъм*



а)

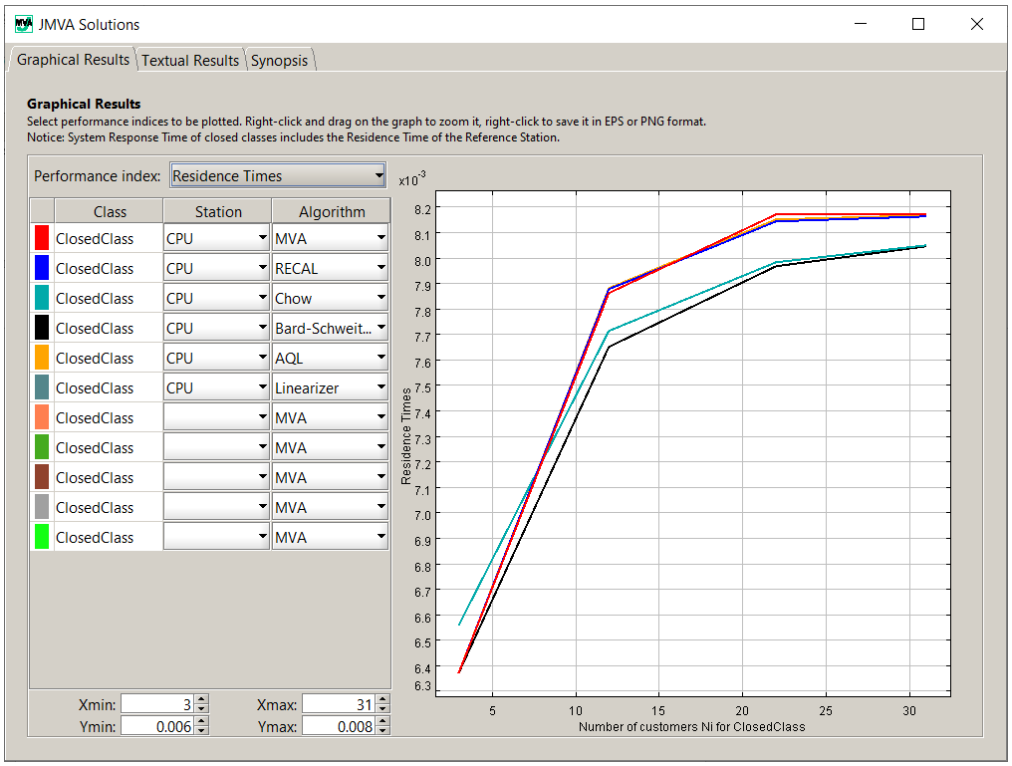


б)

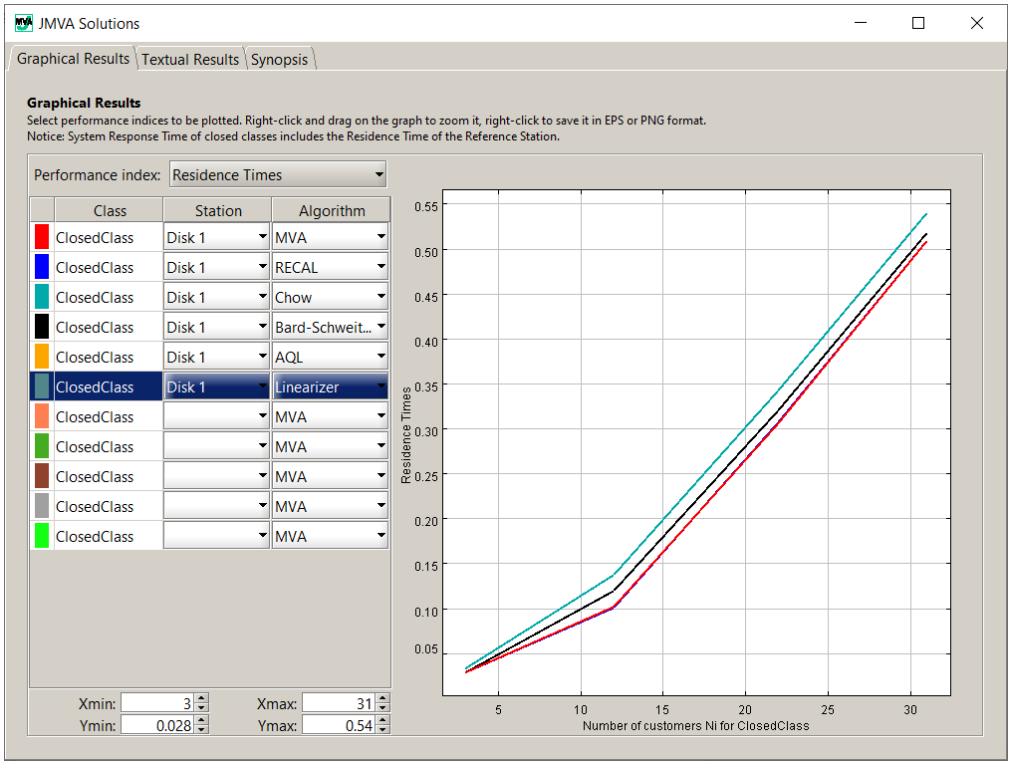


в)

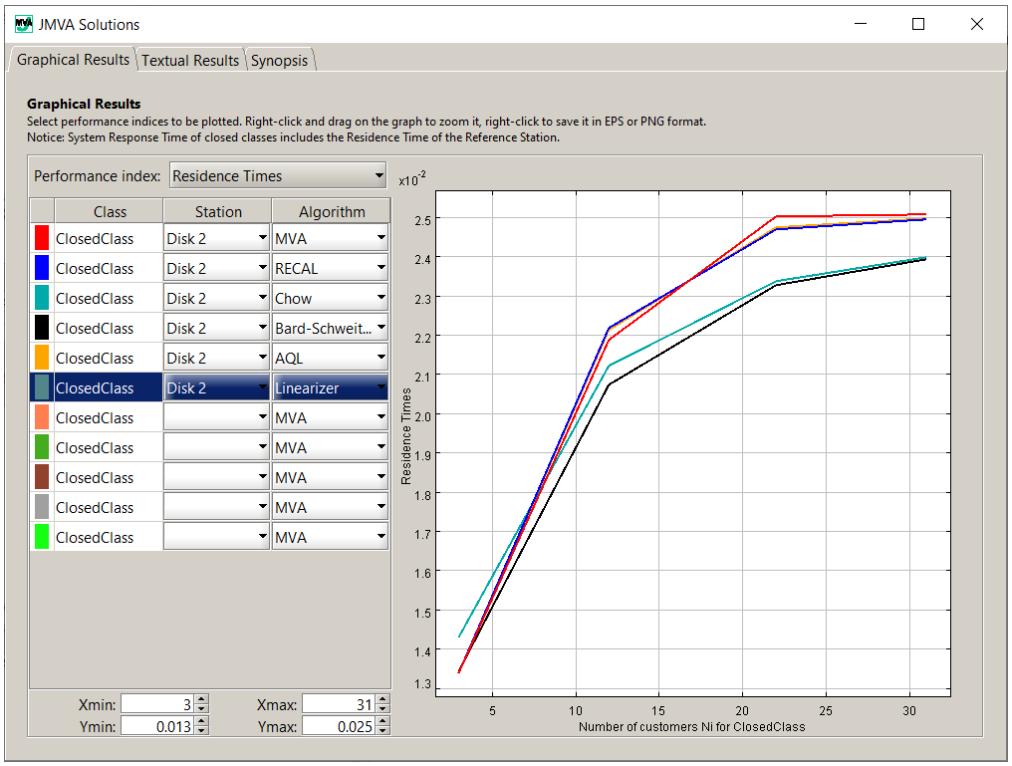
Фиг. 1.19. *„Пропускателна способност“ при “Exact” и “Approximate” алгоритъм за обслужваща станция а)“CPU”, б) “Disk 1” и в)”Disk 2”*


JMVA Solutions
Graphical Results
Textual Results
Synopsis
Graphical Results
Select performance indices to be plotted. Right-click and drag on the graph to zoom it, right-click to save it in EPS or PNG format.
Notice: System Response Time of closed classes includes the Residence Time of the Reference Station.
Performance index: Residence Times
Class
Station
Algorithm
ClosedClass CPU MVA
ClosedClass CPU RECAL
ClosedClass CPU Chow
ClosedClass CPU Bard-Schweit...
ClosedClass CPU AQL
ClosedClass CPU Linearizer
ClosedClass MVA
ClosedClass MVA
ClosedClass MVA
ClosedClass MVA
ClosedClass MVA
Xmin: 3
Xmax: 31
Ymin: 0.006
Ymax: 0.008
x10-3
Residence Times
Number of customers Ni for ClosedClass

а)

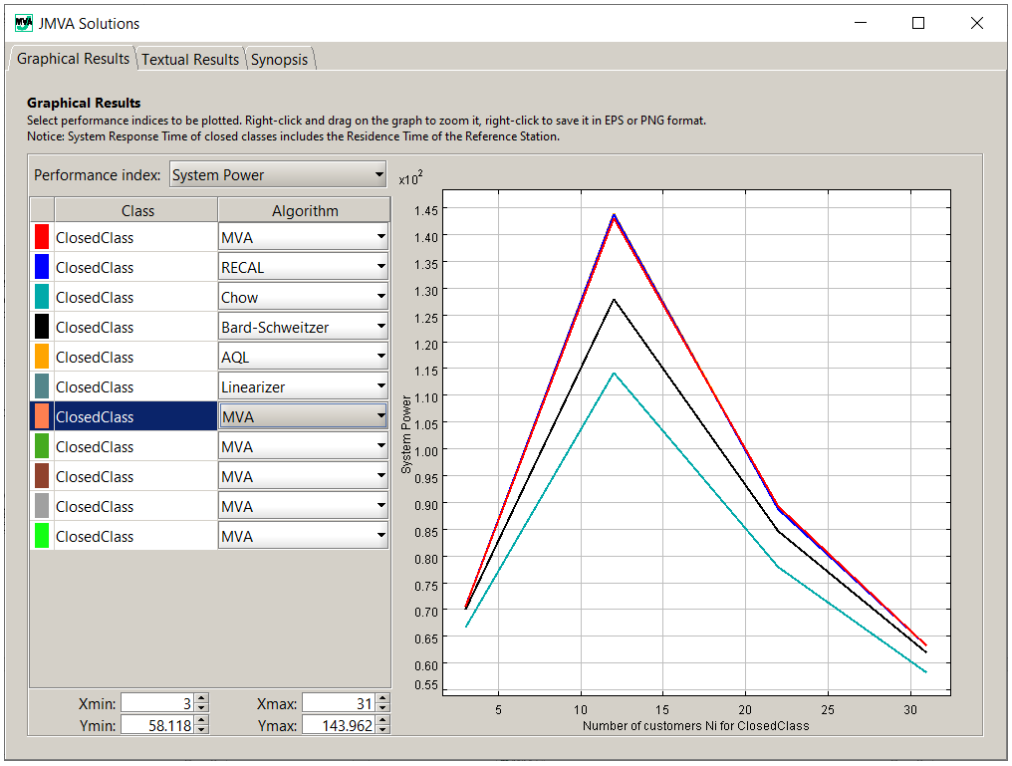

JMVA Solutions
Graphical Results
Textual Results
Synopsis
Graphical Results
Select performance indices to be plotted. Right-click and drag on the graph to zoom it, right-click to save it in EPS or PNG format.
Notice: System Response Time of closed classes includes the Residence Time of the Reference Station.
Performance index: Residence Times
Class
Station
Algorithm
ClosedClass Disk 1 MVA
ClosedClass Disk 1 RECAL
ClosedClass Disk 1 Chow
ClosedClass Disk 1 Bard-Schweit...
ClosedClass Disk 1 AQL
ClosedClass Disk 1 Linearizer
ClosedClass MVA
ClosedClass MVA
ClosedClass MVA
ClosedClass MVA
ClosedClass MVA
Xmin: 3
Xmax: 31
Ymin: 0.028
Ymax: 0.54
Residence Times
Number of customers Ni for ClosedClass

б)



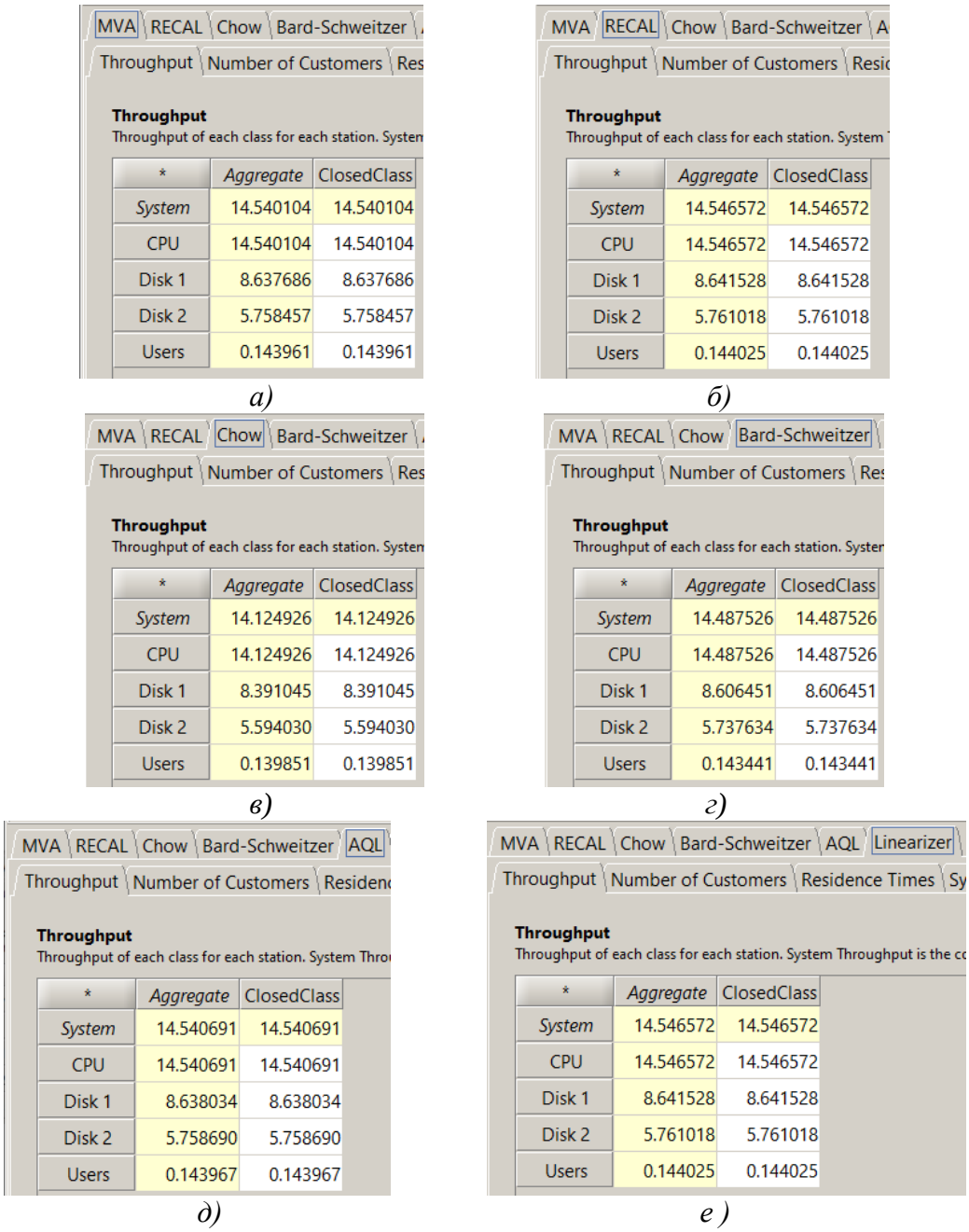
в)

Фиг. 1.20. *„Сумарно време за престой в опашката и обслужване в рамките на всички постъпвания на заявки“ при “Exact” и “Approximate” алгоритми за обслужваща станция а)“CPU”, б) “Disk 1” и в)”Disk 2”*

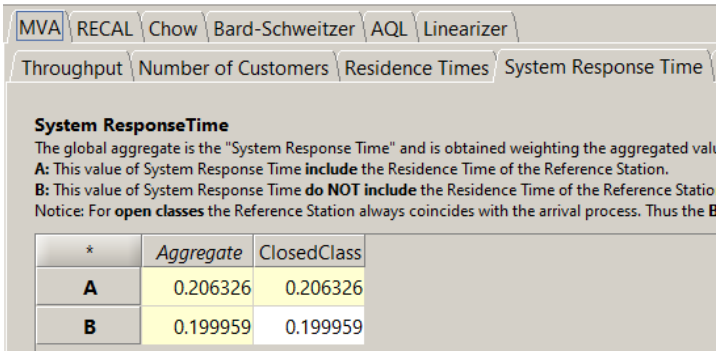


Фиг. 1.21. *„Отношение „Пропускателна способност“ към „Сумарно време за чакане в опашката и обслужване при еднократно постъпване на заявки““ при “Exact” и “Approximate” моделиращи алгоритми*

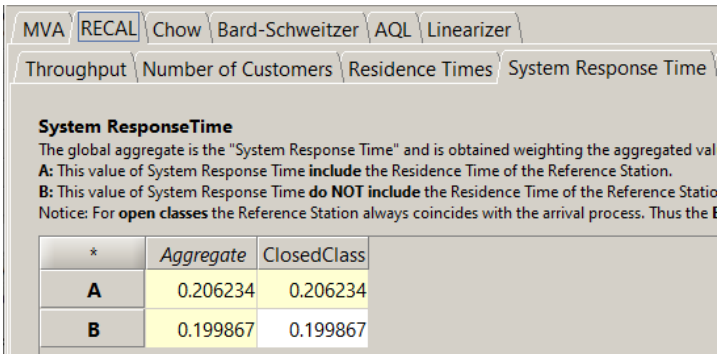
- На следващ етап е необходимо да се проведе задълбочен анализ и да се извърши избор на най-подходяща категория технически подход - “Exact”, “Approximate” и съответен тип алгоритъм - ***MVA, RECAL, Chow, Bard-Schweitzer, AQL, Linearizer, гарантиращ оптимални нива на изследваните индекси на производителност при моделиране на мрежовата структура.*** За целта могат да бъдат използвани **калкулираните начални или минимални стойности** на изследваните трафични параметри за цялостната система и отделно при всяка структурна единица на модела от фиг. 1.22 до фиг. 1.24;



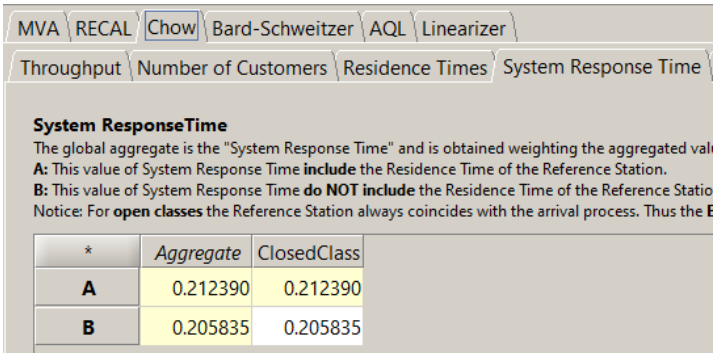
Фиг. 1.22. *Минимални нива на „Пропускателната способност“ за системата и отделните структурни станции за а) “MVA”, б) “RECAL”, в) “Chow”, г) “Bard-Schweitzer”, д) “AQL” и е) “Linearizer”*


MVA
RECAL
Chow
Bard-Schweitzer
AQL
Linearizer
Throughput
Number of Customers
Residence Times
System Response Time
System ResponseTime
The global aggregate is the "System Response Time" and is obtained weighting the aggregated val
A: This value of System Response Time include the Residence Time of the Reference Station.
B: This value of System Response Time do NOT include the Residence Time of the Reference Statio
Notice: For open classes the Reference Station always coincides with the arrival process. Thus the
* Aggregate ClosedClass
A 0.206326 0.206326
B 0.199959 0.199959

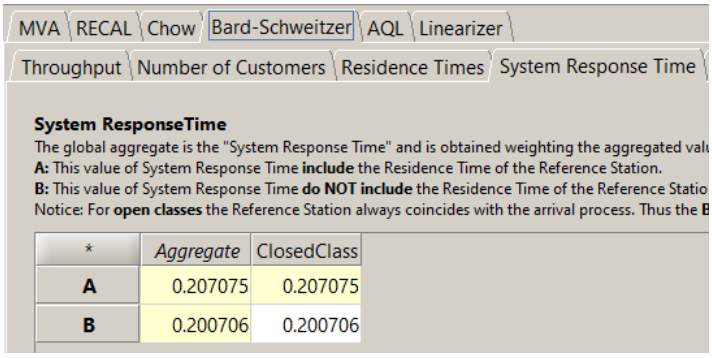
а)


MVA
RECAL
Chow
Bard-Schweitzer
AQL
Linearizer
Throughput
Number of Customers
Residence Times
System Response Time
System ResponseTime
The global aggregate is the "System Response Time" and is obtained weighting the aggregated val
A: This value of System Response Time include the Residence Time of the Reference Station.
B: This value of System Response Time do NOT include the Residence Time of the Reference Statio
Notice: For open classes the Reference Station always coincides with the arrival process. Thus the
* Aggregate ClosedClass
A 0.206234 0.206234
B 0.199867 0.199867

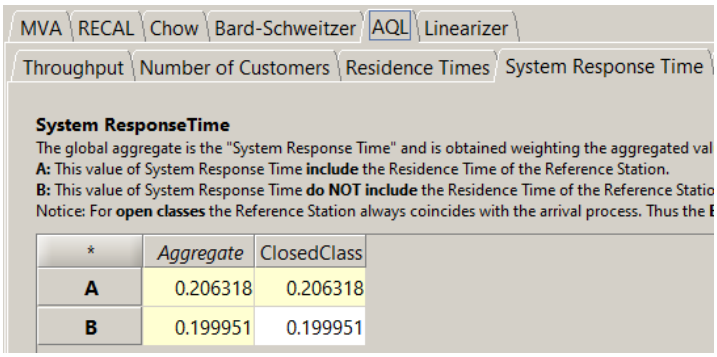
б)


MVA
RECAL
Chow
Bard-Schweitzer
AQL
Linearizer
Throughput
Number of Customers
Residence Times
System Response Time
System ResponseTime
The global aggregate is the "System Response Time" and is obtained weighting the aggregated val
A: This value of System Response Time include the Residence Time of the Reference Station.
B: This value of System Response Time do NOT include the Residence Time of the Reference Statio
Notice: For open classes the Reference Station always coincides with the arrival process. Thus the
* Aggregate ClosedClass
A 0.212390 0.212390
B 0.205835 0.205835

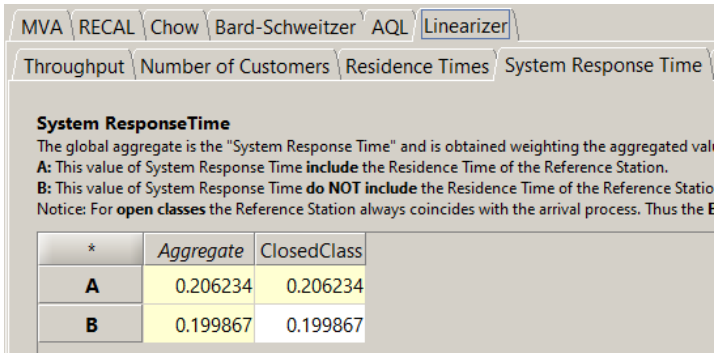
в)


MVA
RECAL
Chow
Bard-Schweitzer
AQL
Linearizer
Throughput
Number of Customers
Residence Times
System Response Time
System ResponseTime
The global aggregate is the "System Response Time" and is obtained weighting the aggregated val
A: This value of System Response Time include the Residence Time of the Reference Station.
B: This value of System Response Time do NOT include the Residence Time of the Reference Statio
Notice: For open classes the Reference Station always coincides with the arrival process. Thus the
* Aggregate ClosedClass
A 0.207075 0.207075
B 0.200706 0.200706

г)

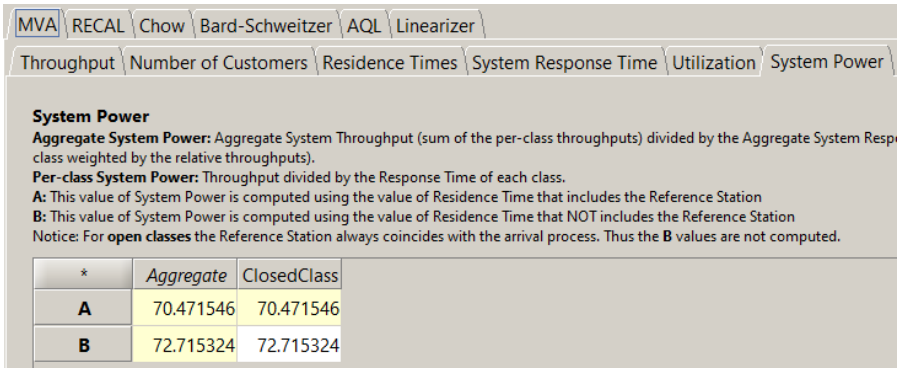


д)

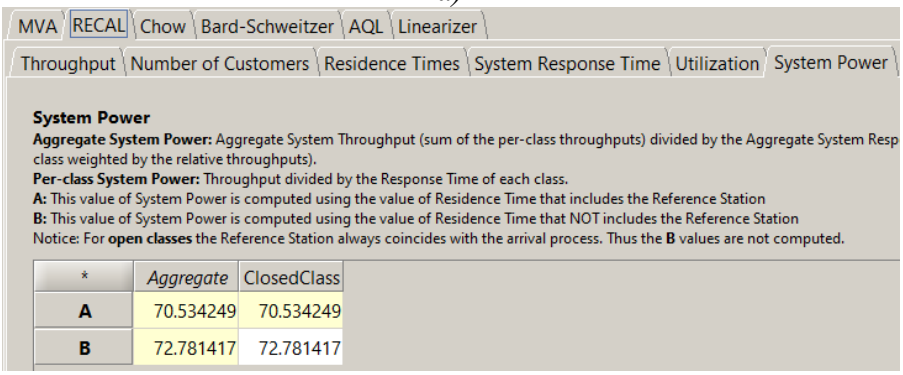


е)

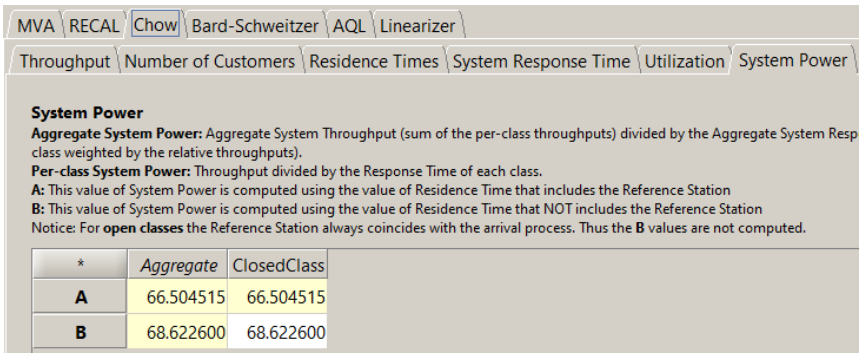
Фиг. 1.23. *Минимални нива на „Системното комплексно време за чакане и обслужване при единично постъпване“ с включване и изключване на „Сумарното време за чакане и обслужване в рамките на всички постъпвания на заявки“ при референтна станция “CPU” за алгоритми а) “MVA”, б) “RECAL”, в) “Ghow”, г) “Bard-Schweitzer”, д) “AQL” и е) “Linearizer”*



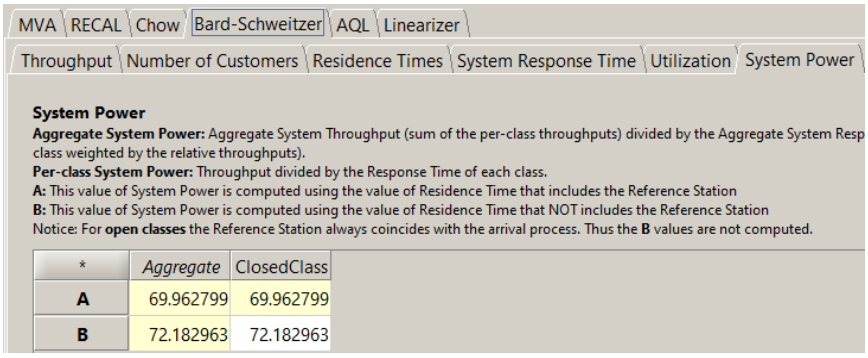
а)



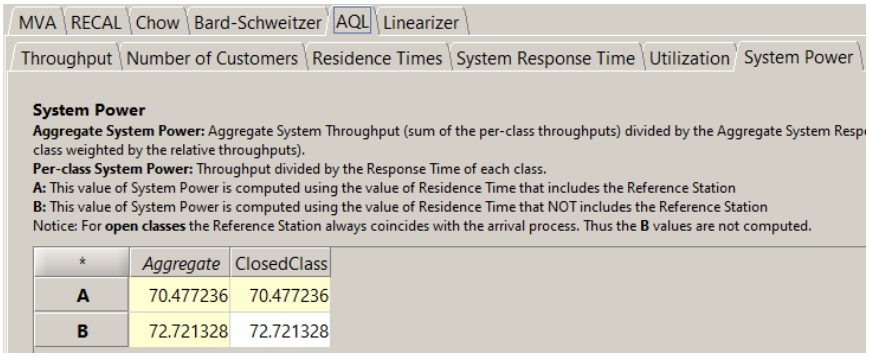
б)



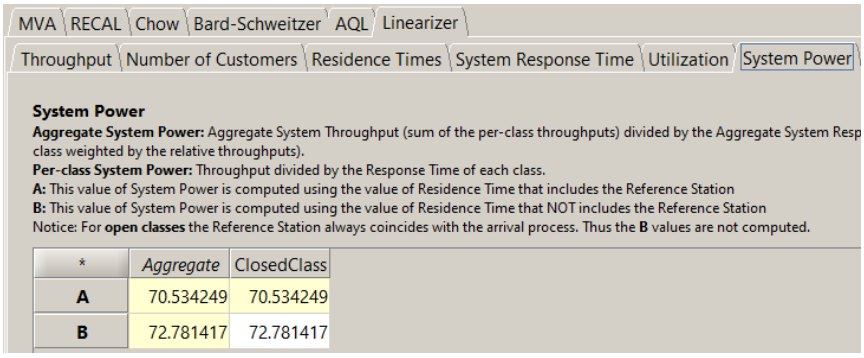
в)



г)

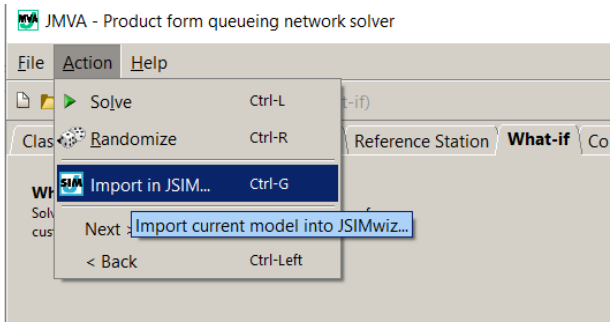


д)

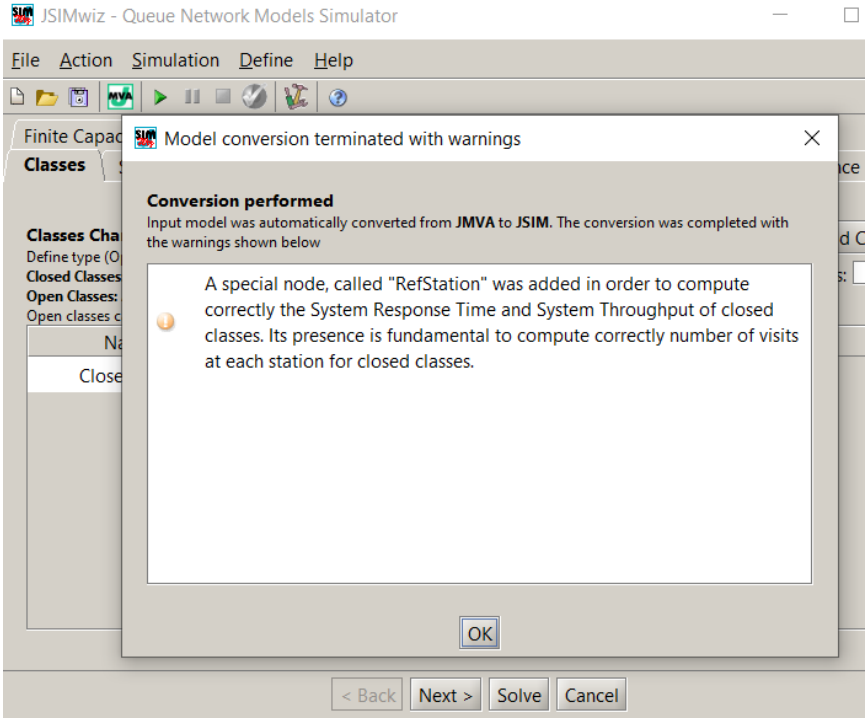


е)

Фиг. 1.24. *Минимални нива на „Параметричното отношение “System Throughput/System Response Time” при единично постъпване“ с включване и изключване на „Сумарното време за чакане и обслужване в рамките на всички постъпвания на заявки“ при референтна станция “CPU” за алгоритми а) “MVA”, б) “RECAL”, в) “Chow”, г) “Bard-Schweitzer”, д) “AQL” и е) “Linearizer”*



а)



б)

Фиг. 1.25. *Избор на а) "Import to JSIMwiz" и б) изглед след процедурата по импортиране в симулатора на мрежи с опашкова организация*

- Импортиране на мрежовия модел с опашкова организация в средата на симулатор JSIMGraph, показано на фиг. 1.25, и прилагане на допълнителни процедури по диагностика съобразно функционалните възможности и особености на средата.

1.4. Контролни въпроси

1. Какви категории алгоритми могат да се използват при имитационното моделиране на телетрафични системи с опашки?
2. Кои са основните индекси, свързвани с производителността и ефективността на обработката на потребителските заявки?
3. Кой индекс на производителността изразява заетостта на обслужващите структурни звена?
4. Кое съотношение определя изискването за оптималност, удовлетворяващо постигането на най-добро бързодействие и надеждност при обслужването на системни потребители?

Лабораторно упражнение № 2

ИЗСЛЕДВАНЕ НА МОДУЛАЦИОННИ ПРОЦЕСИ И ФАДИНГ ЕФЕКТ В КОМУНИКАЦИИТЕ ЧРЕЗ MATLAB СОФТУЕР

2.1. Цел на упражнението

Да се запознаят студентите с аналитичните инструменти и графични техники за представяне на основни типове модулации и изследване на ефекта на Rayleigh Fading при предаването на Binary DPSK модулиран сигнал в комуникационните канали за връзка.

2.2. Теоретична постановка.

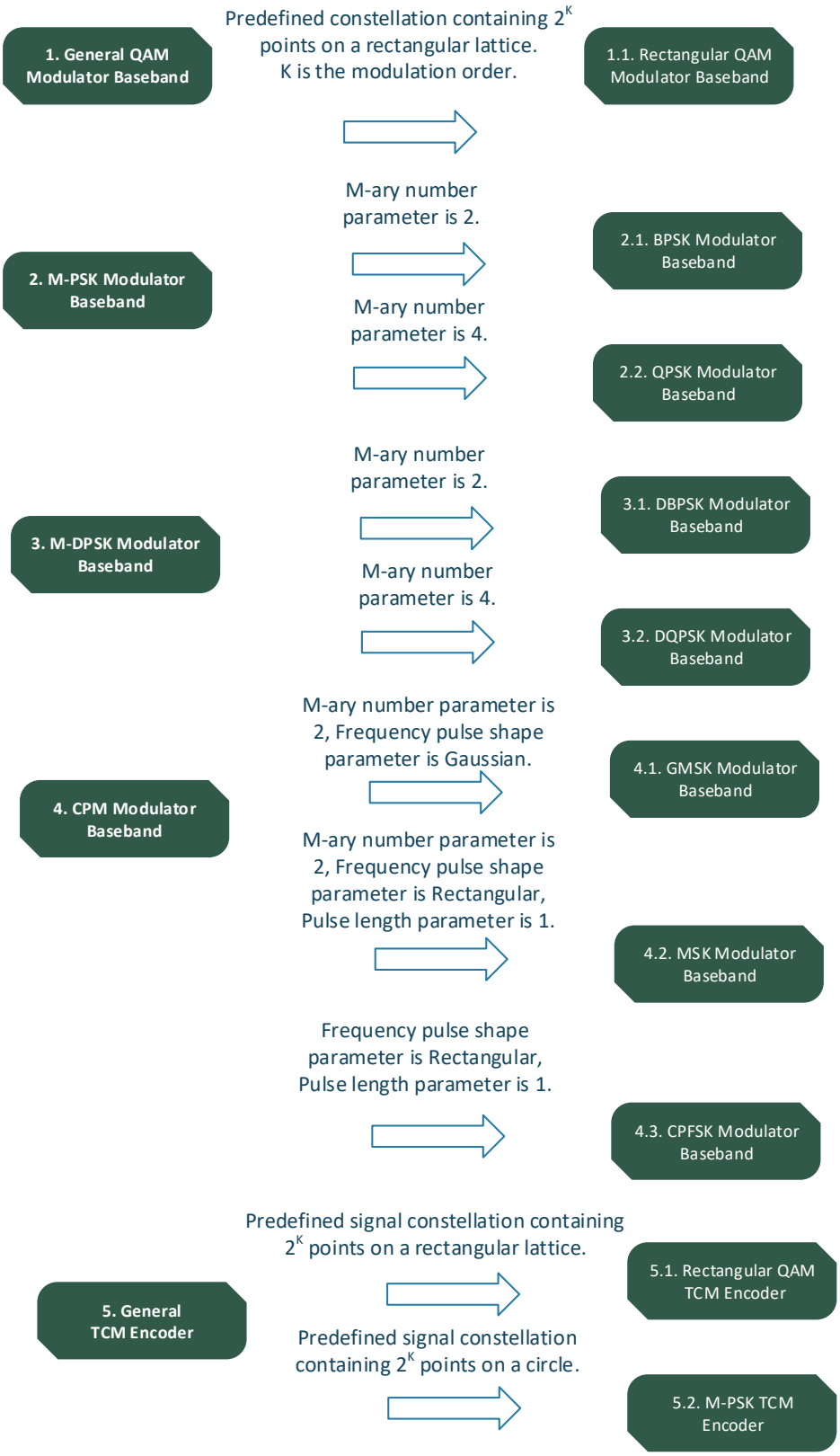
Подходи за цифрова модулация

Изборът на подходящ тип модулация в комуникационните системи има важно значение относно ефективното използване на ресурсите на честотната лента. В това отношение, системите с използване на цифрово модулирани сигнали се отличават с по-намалено влияние на смущенията, изкривявания във формата на сигналите, нелинейността, шумовете и др. В редица случаи предимствата на цифровата пред аналоговата модулация се състоят в:

- Повишен капацитет при предаването на данни;
- Висока ефективност на честотната лента;
- Гъвкавост на сигналите;
- Понижена чувствителност спрямо случайни въздействия и ефекти;
- По-висока интензивност на сигналите;
- Повишено съотношение „сигнал-шум“;
- Икономическа ефективност при проектирането на цифровите комутационни системи;
- Гарантиране на сигурност и надеждност на данните чрез възможност за прилагане на ефективни подходи за информационно криптиране и декриптиране [3, 4].

Фиг. 2.1. *Класификация на методите за модулиране на цифрови сигнали*

Фигура 2.1 обобщава методите за модулиране на цифровите сигнали, приложими в комуникациите, докато на фиг. 2.2 са дадени основните и еквивалентните им специфични функционални блокове за модулация в MATLAB среда.



Фиг. 2.2. *Основни и еквивалентни специфични блокове за модулация в MATLAB*

Едни от най-разпространените видове модулации при предаване на сигнали в комуникационните системи са:

- PSK (съкратено от Phase-Shift Keying), която се разглежда като форма на модулация, която представя цифровите данни, като промени във фазата на носещото трептене. Някои от нейните основни характеристики са:

  - ✓ Амплитудата и честотата остават постоянни, докато фазата на сигнала се променя;

  - ✓ Промяната на фазовото състояние е лесно за регистриране като за целта най-често приеманият сигнал се сравнява с референтен сигнал.

  Съществуват различни PSK модификаци, които са широко разпространени в съвременните комуникации. Могат да бъдат споменати следните варианти:

  - ✓ BPSK или Binary Phase-Shift Keying;

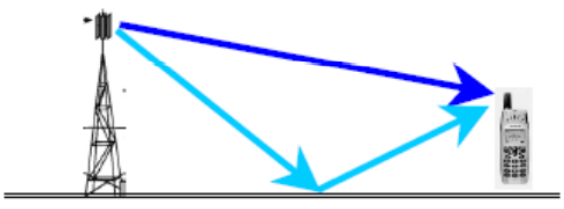
  - ✓ QPSK или Quadrature Phase-Shift Keying;

  - ✓ OQPSK или Offset Quadrature Phase-Shift Keying;

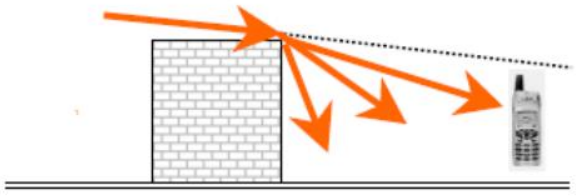
  - ✓ GMSK или Gaussian Minimum Shift Keying.

- QAM (съкратено от Quadrature amplitude modulation), прилагаща се за значително подобряване на шумоустойчивостта на комуникационните сигнали при едновременно въздействие върху амплитудата и фазата на сигнала на носещия сигнал. По този начин се получава т. нар. m-QAM (m – цяло число; броят на различните амплитудни и фазови състояния на сигнала), при което се предават едновременно $\log_2 m$ бита. Например за 16-QAM модулация е налице едновременно предаване на 4 бита в 16 различни състояния. Този тип модулация до известна степен наподобява вида на 16-PSK, но се отличава с по-добра шумозащитеност, тъй като съседните фазови състояния се различават и по амплитуда. 16-QAM модулацията се използва в съвременните 4G LTE, WiMAX и други системи, когато нивото на сигнала е високо [3, 4].

Един от най-честите ефекти при предаването на сигнали в мобилните комуникации, резултат от физичните явления „интерференция“ – фиг. 2.3.а), или „дифракция“ – фиг. 2.3.б), е така нареченият „Fading“. Действието на ефекта – „бърз фадинг“ вследствие от „интерференция“ и „бавен фадинг“ резултат от „дифракция“, се изразява в промяна на нивото на сигнала. Действието на затихването и при двата варианта на ефекта са показан на фиг. 2.4.

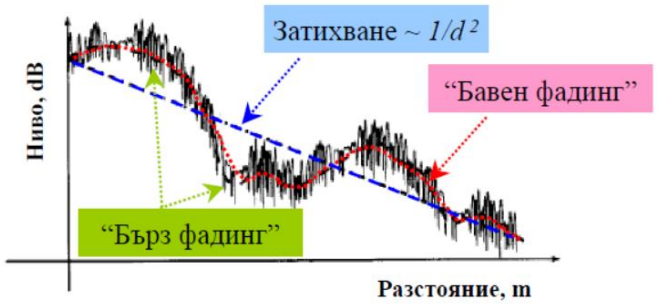


а)



б)

Фиг. 2.3. *Физични явления а) „интерференция“ и б) „дифракция“*



Фиг. 2.4. *Затихване, бърз и бавен фадинг*

По отношение на илюстрираните ефекти на фиг. 2.4 могат да бъдат направени следните дефиниции [3, 4]:

- Затихване от разстоянието (6 dB или 4 пъти на всяко удвояване на разстоянието). Това затихване е неизбежно и е свързано с разширението на фронта на вълната;

- Бавен фадинг (засенчване): максимуми през 10-100 m и затихване от максимум до минимум ~4-10 dB; свързан е с дифракция от ръбовете на големи препятствия с размери D » λ (сгради, хълмове и други теренни препятствия);

- Бърз фадинг: максимуми през λ/2 (т. е. ~ 17 cm при 900 MHz) и голямо затихване от максимум до минимум до 30-40 dB; свързан е с разсейването на сигнала от малки обекти с размери D ~ λ и възникването на “многолъчевост”.

2.3. Последователност на работа и задачи за изпълнение

2.3.1. Визуализация и изследване на модулирани сигнали чрез диаграми на констелация в MATLAB среда.

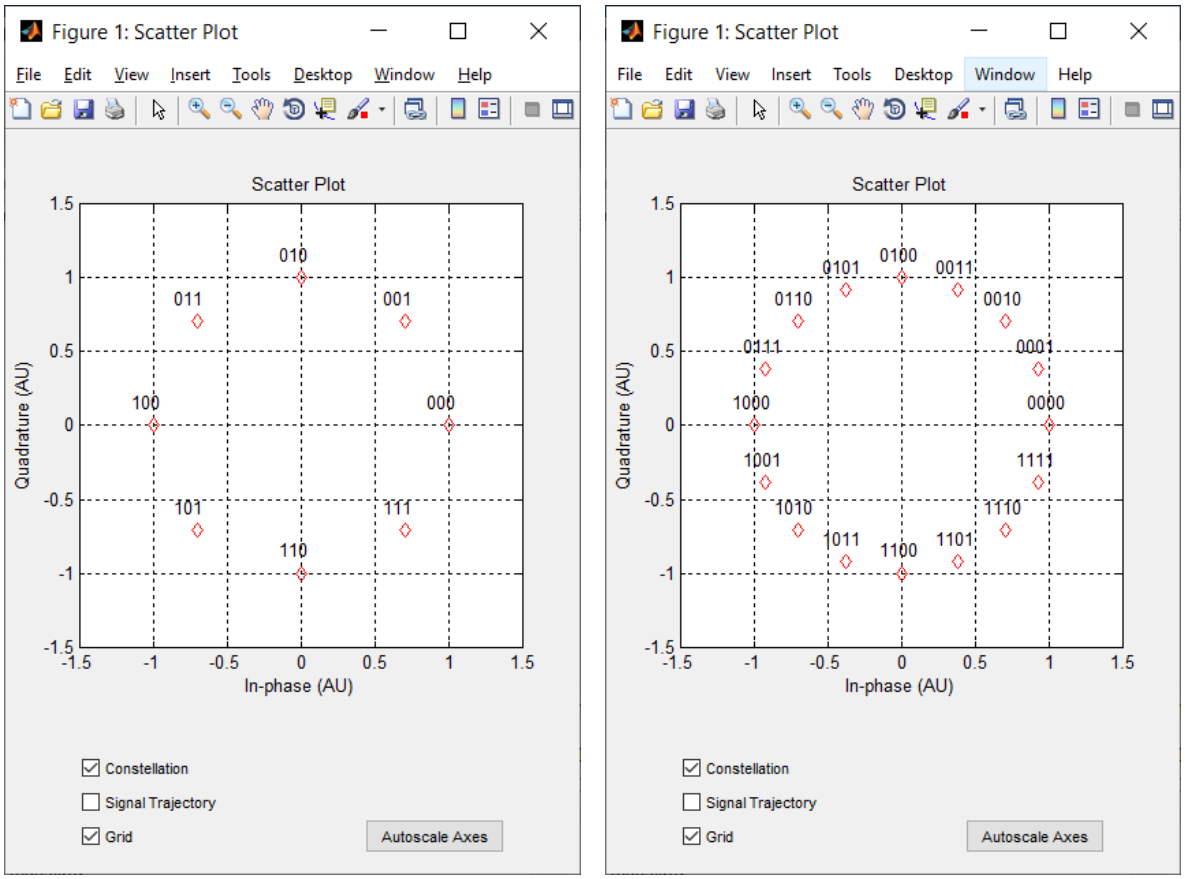
Един от графичните инструменти за визуализация и изследване на изходните модулирни сигнали в процесите на създаване и синтез на PSK и QAM модулатори са така нар. „констелационни диаграми“. Относно целите на изследването е разгледана възможността за генериране на различни типове PSK и QAM констелации в средата MATLAB. Последователните процеси на създаване на всички възможни входове на съответеният вид модулатор, прилагане на подходяща модулираща функция и извеждане на изходния модулиран сигнал се базират на специализирани функции, формиращи съответен script.

- ***Изследвания, свързани с PSK модулация***

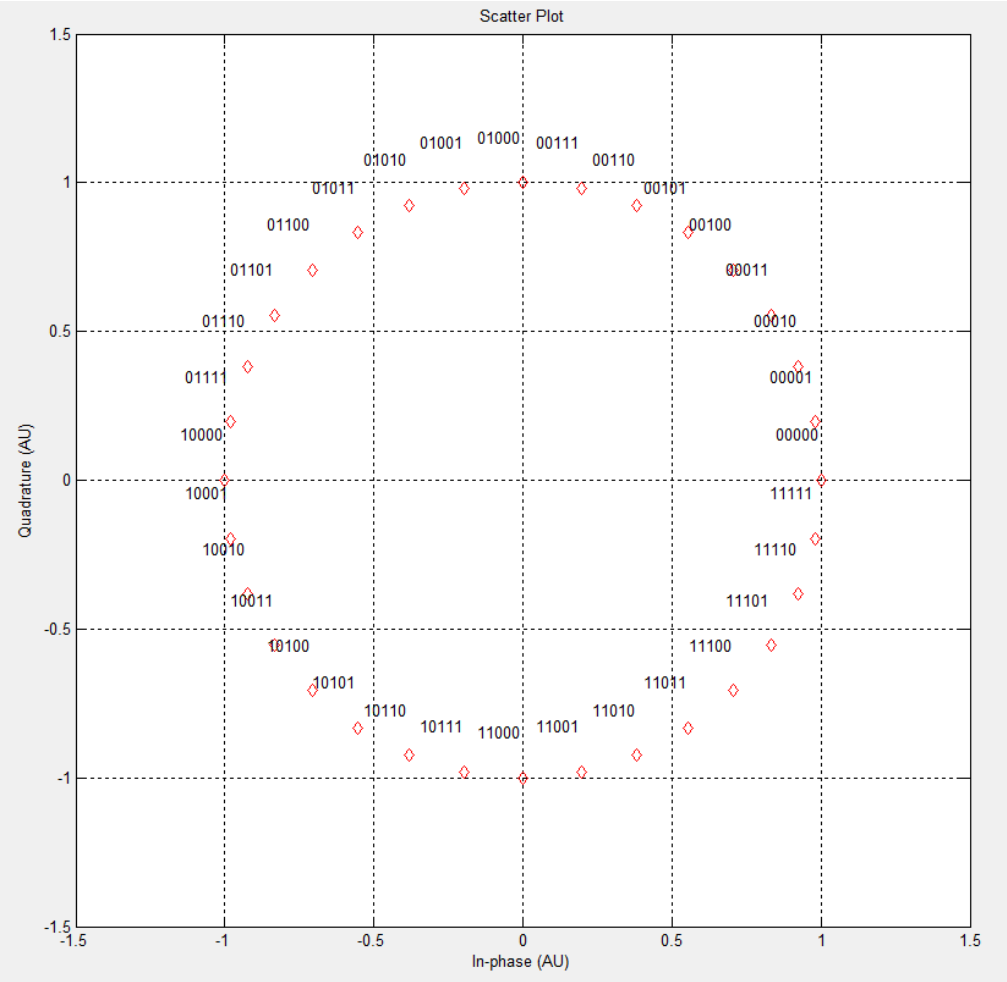
Фигура 2.5 представя сорс код, въведен в командния прозорец, посредством който се извършва последователно създаване на 8-PSK, 16-PSK и 32-PSK модулации и генериране на сигналните диаграми на констелация на изходните модулирани сигнали, показани на фиг. 2.6.

```
>> hMod = modem.pskmod(16);
scatterPlot = commscope.ScatterPlot('SamplesPerSymbol',1,'Constellation',hMod.Constellation);

scatterPlot.PlotSettings.Constellation = 'on';
scatterPlot.PlotSettings.ConstellationStyle = 'rd';

hold on;
k=log2(hMod.M);
for jj=1:hMod.M
        text(real(hMod.Constellation(jj))-0.15,imag(hMod.Constellation(jj))+0.15,dec2base(hMod.SymbolMapping(jj),2,k));
end
hold off;
```

Фиг. 2.5. *Script за създаване на 8-PSK, 16-PSK и 32-PSK модулации и визуализация на 8-PSK, 16-PSK и 32-PSK констелации*



а) *б)*



в)

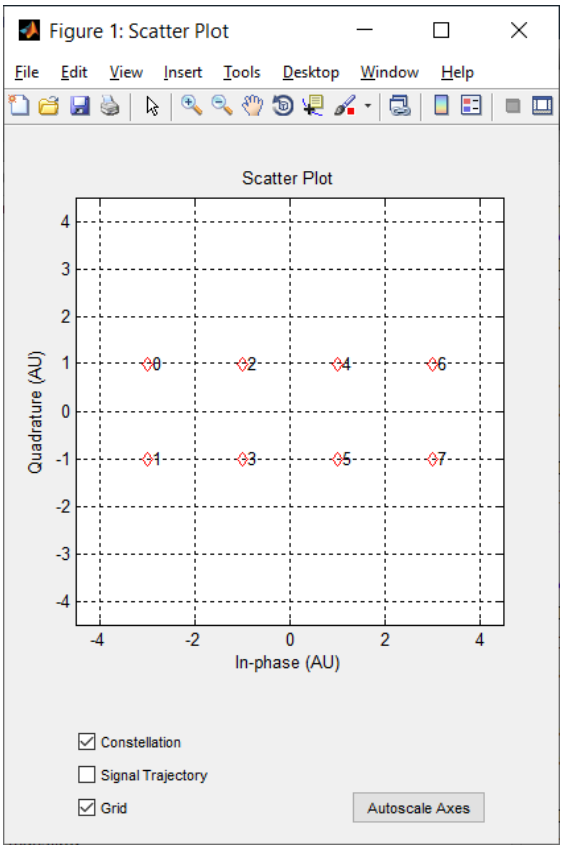
Фиг. 2.6. *Диаграми на констелации на а) 8-PSK, б) 16-PSK и в) 32-PSK модулирани сигнали*

- ***Изследвания, насочени с QAM модулация***

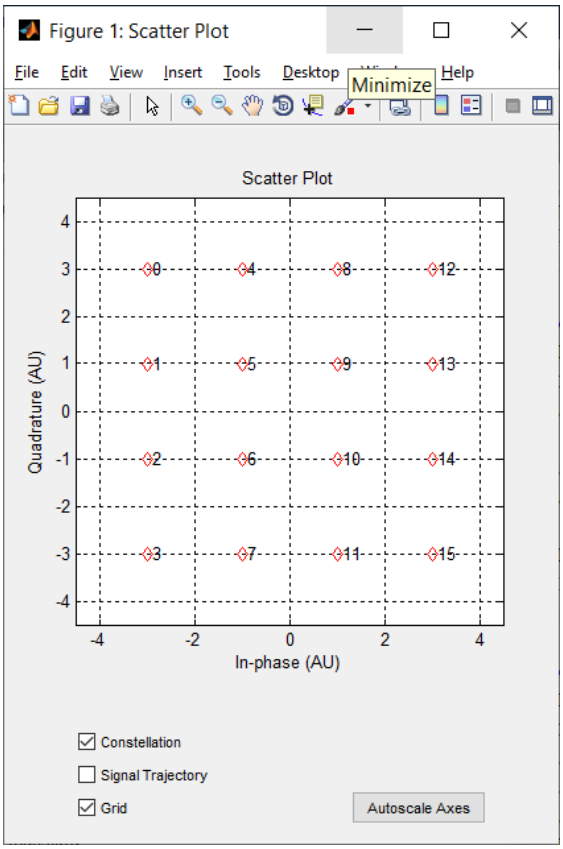
По-долу е показан script кодът за формиране на 8-QAM, 16-QAM, 32-QAM и 64-QAM типове модулация при пикова мощност, равняваща се на 1 W (фиг. 2.7). Получените изходни 8-QAM, 16-QAM, 32-QAM и 64-QAM диграми на констелация са дадени на фиг. 2.8. В друг вариант на процедурата е предвидено използването на Код на Грей, чрез който се създават 8-QAM, 16-QAM, 32-QAM и 64-QAM Gray кодер модулации, след което по аналогичен начин могат да бъдат визулизирани конкретните диаграми на констелация. Дейностите са онагледени на фиг. 2.9 и фиг. 2.10.

```
>> hMod = modem.qammod(32);
scatterPlot = commscope.ScatterPlot('SamplesPerSymbol',1,'Constellation',hMod.Constellation);
scatterPlot.PlotSettings.Constellation = 'on';
scatterPlot.PlotSettings.ConstellationStyle = 'rd';
hold on;
for jj=1:hMod.M
   text(real(hMod.Constellation(jj)),imag(hMod.Constellation(jj)),[' ' num2str(hMod.SymbolMapping(jj))]);
end
hold off;
```

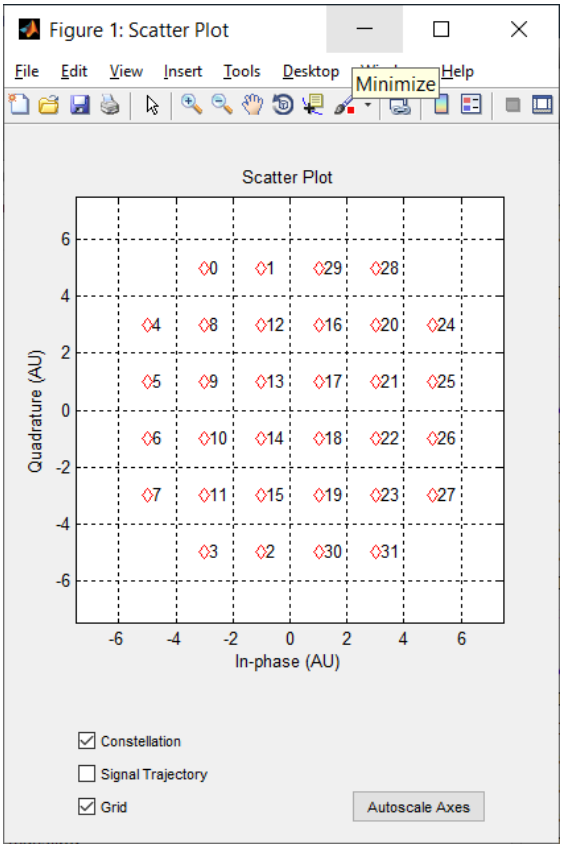
Фиг. 2.7. *Script за създаване на 8-QAM, 16-QAM, 32-QAM и 64-QAM модулации и генериране на 8-QAM, 16-QAM, 32-QAM и 64-QAM констелации*



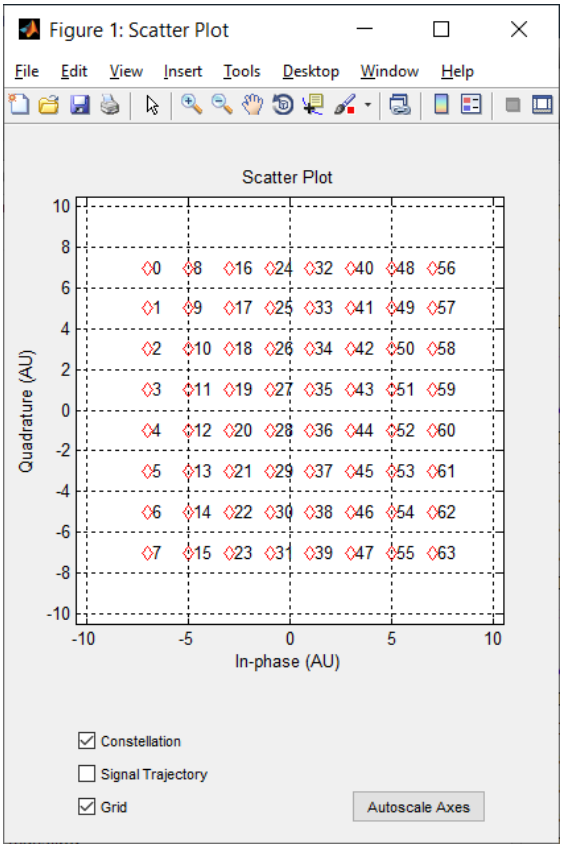
а)



б)



в)



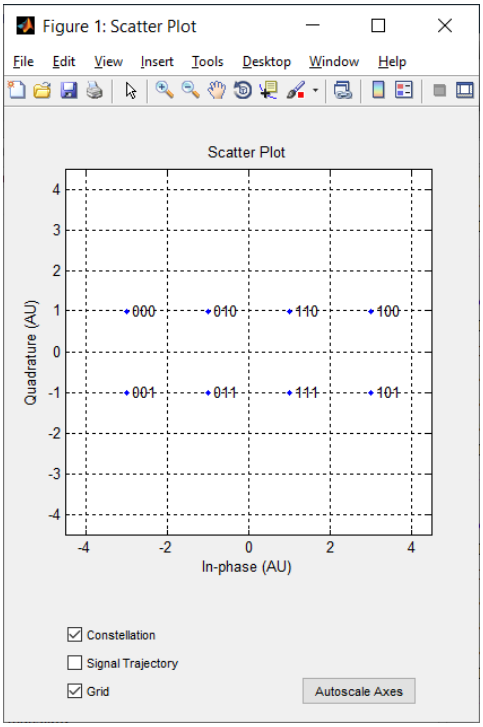
г)

Фиг. 2.8. *Констелационни диаграми на а) 8-QAM, б) 16-QAM, в) 32-QAM и в) 64-QAM изходни модулирани сигнали*

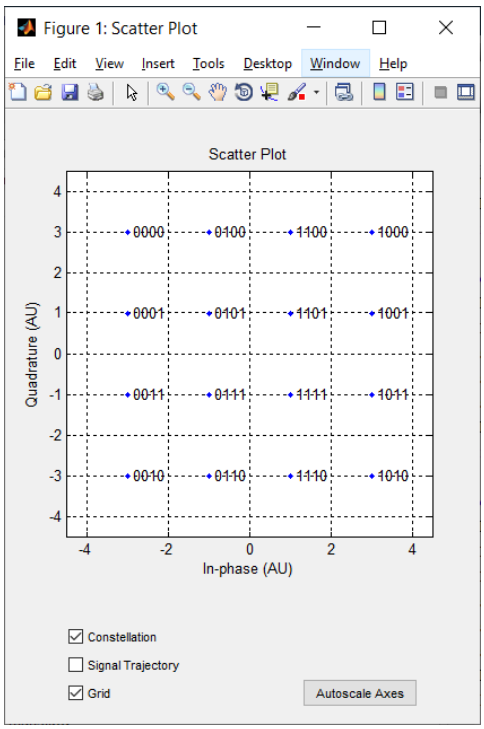
```
>> hMod = modem.qammod('M',16,'SymbolOrder','Gray');
scatterPlot = commscope.ScatterPlot('SamplesPerSymbol',1,'Constellation',hMod.Constellation);
scatterPlot.PlotSettings.Constellation = 'on';
scatterPlot.PlotSettings.ConstellationStyle = '.';
hold on;
k=log2(hMod.M);
for jj=1:hMod.M
        text(real(hMod.Constellation(jj))+0.15,imag(hMod.Constellation(jj)),dec2base(hMod.SymbolMapping(jj),2,k));
end
hold off;
```

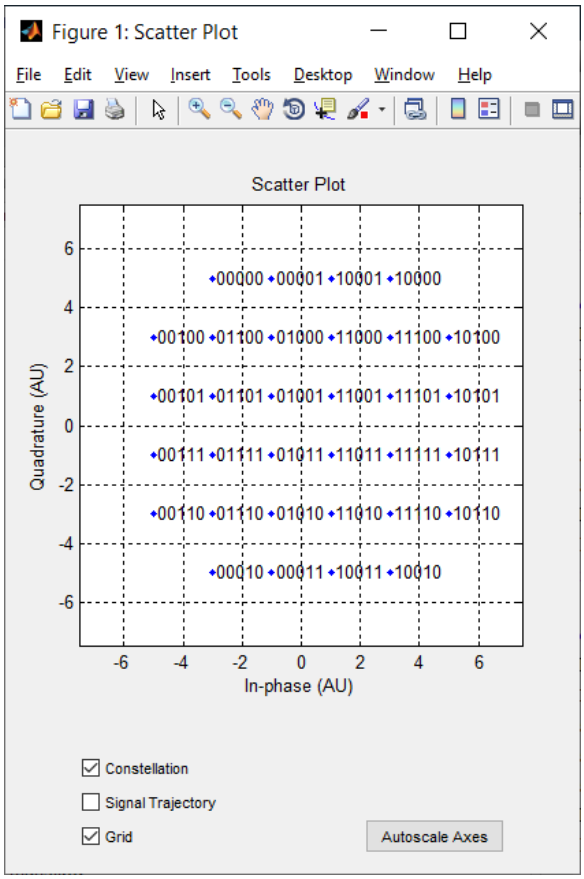
Фиг. 2.9. *Script за създаване на 8-QAM, 16-QAM, 32-QAM и 64-QAM Gray кодер модулации и визуализация на 8-QAM, 16-QAM, 32-QAM и 64-QAM сигнални Gray-coded констелации*

а)

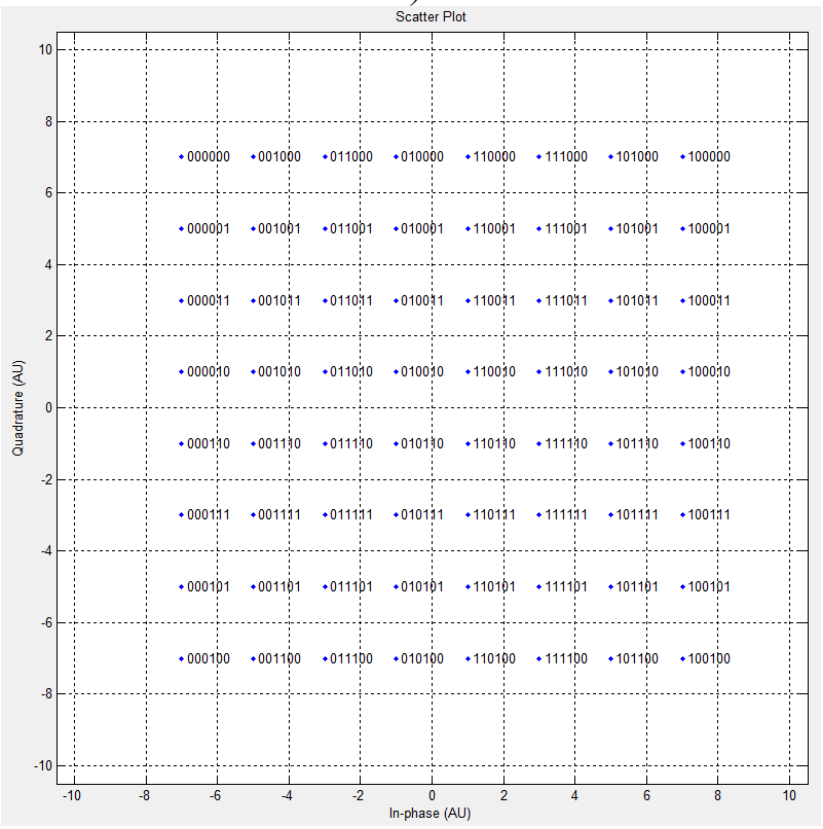


б)





в)



г)

Фиг. 2.10. *Изходни сигнални за: а) 8-QAM, б) 16-QAM, в) 32-QAM и г) 64-QAM Gray-coded констелации*

2.3.2. Изследване на ефекта на Rayleigh Fading канал при предаване на Binary DPSK модулиран сигнал чрез оценка на BER параметъра в MATLAB среда.

Поставя се задачата за *изследване на ефекта на Frequency-Flat Rayleigh Fading канал при предаване на Binary DPSK модулиран сигнал на основата на оценка на BER (Bit Error Ratio) параметъра за различни стойности на критерия SNR (Signal-to-Noise Ratio)*. В тази връзка е съставена последователност от действия, която е препоръчителна при комбиниране на фадинг канала с AWGN (Additive White Gaussian Noise) въздействия, както следва:

- Създаване на Rayleigh фадинг канал за семплиран сигнал при честота 10000 Hz и максимално Доплерово изместване 100 MHz;

```
chan = rayleighchan(1/10000,100);
```

- Дефиниране на реда на DBPSK (Differential Binary Phase Shift Keying) модулация (за конкретния случай $2^{-ри}$ ред);

```
M = 2;
```

- Генериране на Differential Binary Phase Shift Keying модулация;

```
hMod = comm.DBPSKModulator;
```

- Създаване на Differential Binary Phase Shift Keying демодулация;

```
hDemod = comm.DBPSKDemodulator;
```

- Генериране на случаен информационен битов поток;

```
tx = randi([0 M-1],50000,1);
```

- Прилагане на Differential Binary Phase Shift Keying модулация по отношение на информационния сигнал;

```
dpskSig = step(hMod, tx);
```

- Прилагане на създадения фадинг канал към DBPSK модулирания сигнал;

```
fadedSig = filter(chan,dpskSig);
```

- Калкулиране на параметъра Error Rate за различни стойности на параметъра Signal-to-Noise Ratio (фиксиране на SNR граници в dB);

```
SNR = 0:2:20;
numSNR = length(SNR);
berVec = zeros(3, numSNR);
```

- Създаване на системен обект, комбиниращ Additive White Gaussian Noise канал и Error Rate калкулатор. Тук се извършва създаване на AWGN канала, добавяне на Gaussian шум, сигнална демодулация и изчисляване на параметъра Error Rate;

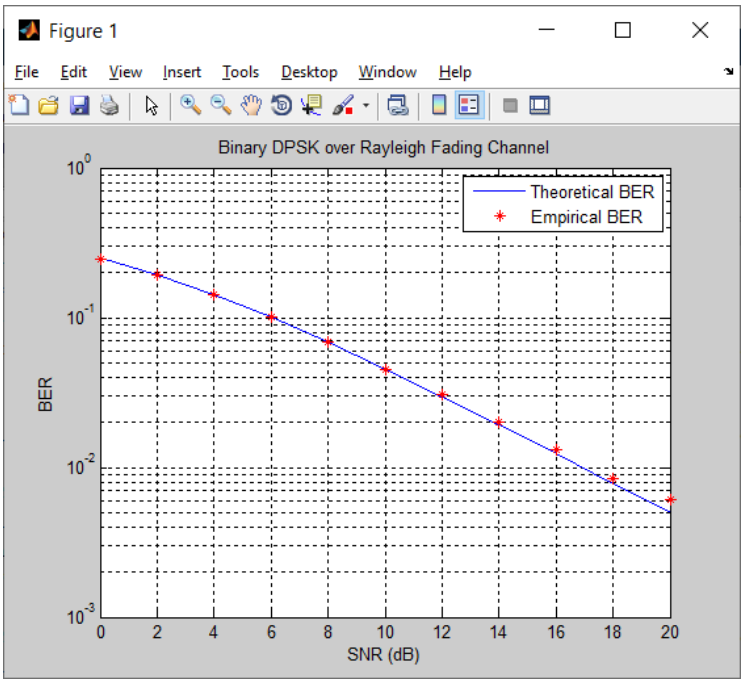
```
hChan = comm.AWGNChannel('NoiseMethod', 'Signal to noise ratio (SNR)');

hErrorCalc = comm.ErrorRate;

for n = 1:numSNR
    hChan.SNR = SNR(n);
rxSig = step(hChan,fadedSig);   % Add Gaussian noise
rx = step(hDemod, rxSig);  % Demodulate
    reset(hErrorCalc)

berVec(:,n) = step(hErrorCalc,tx,rx);
    end
BER = berVec(1,:);
```

- Пресмятане на теоретични стойности на BER (Bit Error Rate) параметъра спрямо обхвата от SNR нива;

```
BERtheory = berfading(SNR,'dpsk',M,1);
```

- Извеждане и сравнителен анализ на теоретичните и симулационните е BER резултати съгласно осцилограмите на фиг. 2.11;

```
semilogy(SNR,BERtheory,'b-',SNR,BER,'r*');
legend('Theoretical BER','Empirical BER');
xlabel('SNR (dB)'); ylabel('BER');
title('Binary DPSK over Rayleigh Fading Channel');
```



а)

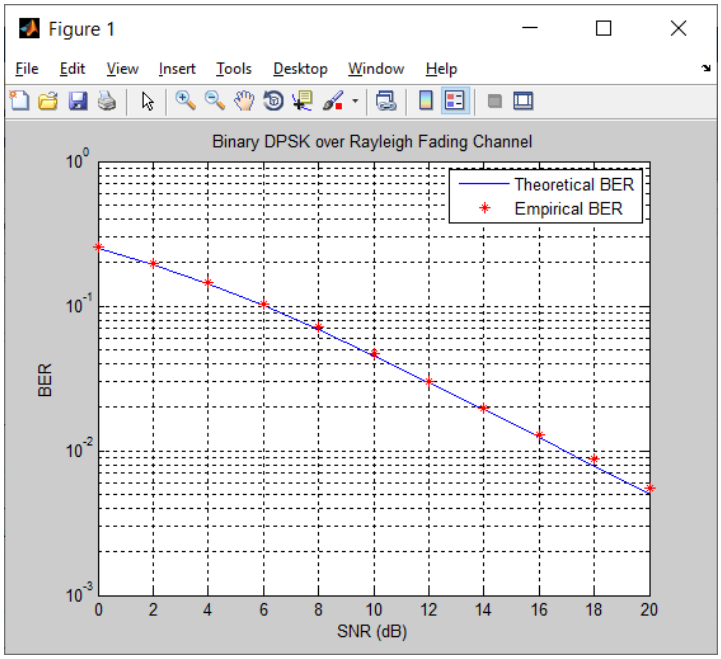


Figure 1
File Edit View Insert Tools Desktop Window Help
Binary DPSK over Rayleigh Fading Channel
Theoretical BER
Empirical BER
BER
SNR (dB)

б)

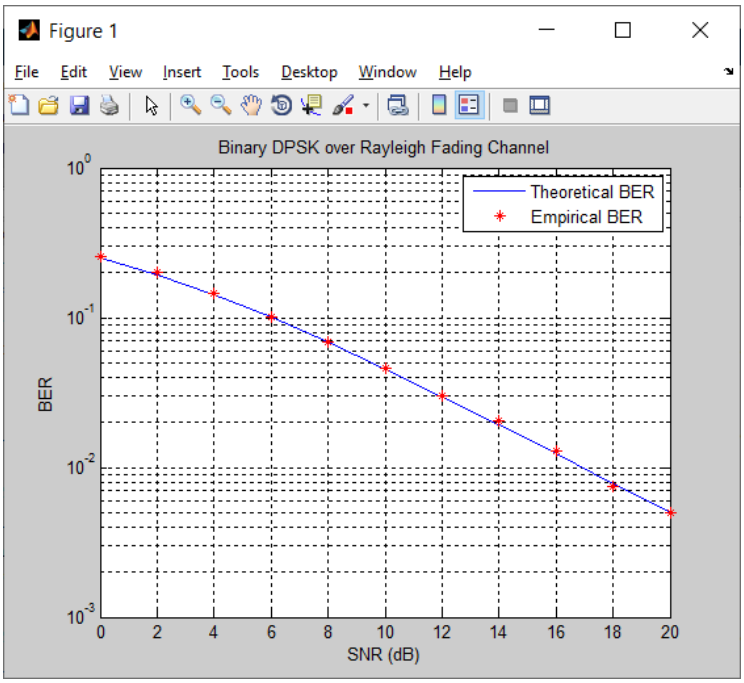
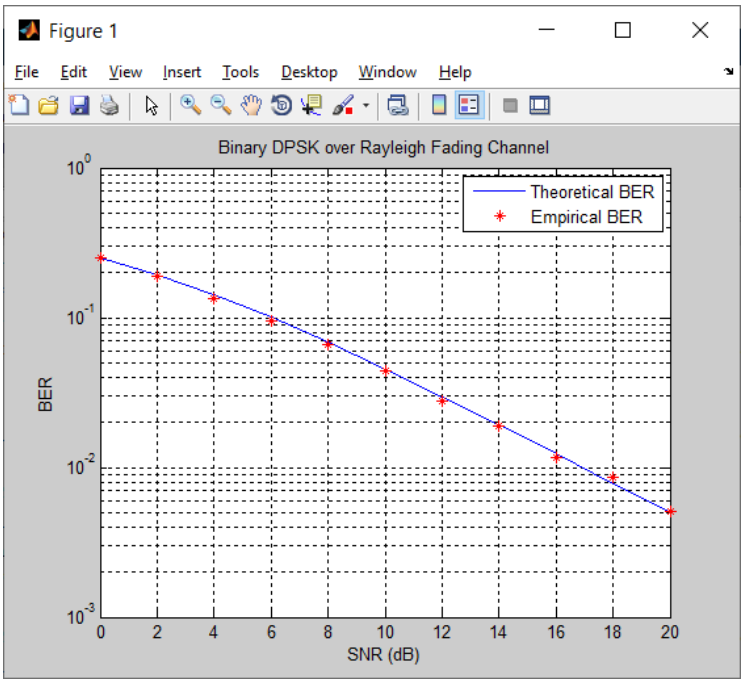


Figure 1
File Edit View Insert Tools Desktop Window Help
Binary DPSK over Rayleigh Fading Channel
Theoretical BER
Empirical BER
BER
SNR (dB)

в)



г)

Фиг. 2.11. *BER параметри при Maximum Doppler shift параметър – а) 100 Hz, б) 110 Hz, в) 120 Hz и г) 130 Hz*

- Повторение на процедурата за различни стойности на Frequency of Signal Sampling / Maximum Doppler shift.

2.4. Контролни въпроси.

1. Какви са предимствата на цифровите в сравнение с аналоговите модулации?
2. В какви категории могат да бъдат разделени методите за модулиране на цифрови сигнали?
3. Какви видове PSK модулации се прилагат в комуникациите?
4. Какво представлява Фадинг ефектът и как се отразява върху качеството на предаваните сигнали?
5. Какво се постига с прилагането на QAM модулация по отношение на носещия сигнал?

Лабораторно упражнение № 3

МОДЕЛИРАНЕ И УПРАВЛЕНИЕ НА TCP И UDP ТРАФИК МЕЖДУ КОМУНИКАЦИОННИ ЗВЕНА ПОСРЕДСТВОМ LABVIEW ВИРТУАЛНИ ИНСТРУМЕНТИ

3.1. Цел на упражнението

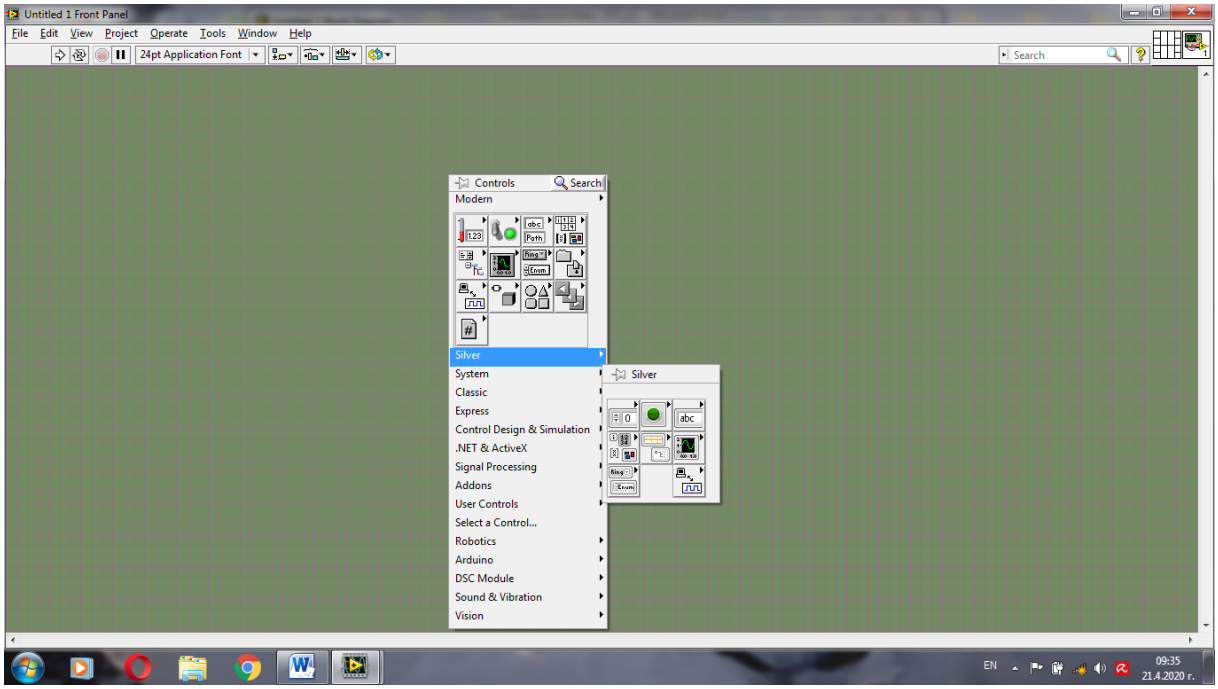
Да се запознаят студентите с възможностите на виртуалната среда LabVIEW за създаване на инструменти тип „сървър/клиент“ и „предавател/приемник“, моделиращи функционалността на транспортните комуникационни мрежови протоколи TCP и UDP.

3.2. Теоретична постановка

Особености и компоненти на програмна среда LabVIEW

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) е платформа за разработка на визуални приложения, използващи графичния език за програмиране G. Софтуерният продукт е разработен от компания National Instruments. Графичната среда е предназначена за създаване на гъвкави приложения за дизайн, управление и тест на минимална себестойност. С LabVIEW потребители от цял свят работят с реални сигнали, анализират данни със съществено значение и обменят резултати чрез подходящи визуални средства. Без значение от степента на опита като програмисти, LabVIEW осигурява създаването на такива приложения бързо и лесно за всички потребители.

Програмите в LabVIEW се наричат виртуални инструменти, тъй като визуално и оперативно физически имитират технически средства като осцилоскопи, мултимери и т.н. Освен това средата разполага с богат набор от инструменти за събиране, анализ, обработка и съхраняване на специфицирани данни и резултати, както и потребителски функци при създаване на кода на програмата. Всяко LabVIEW приложение съдържа два градивни компонента – преден панел и блокова диаграма [5].

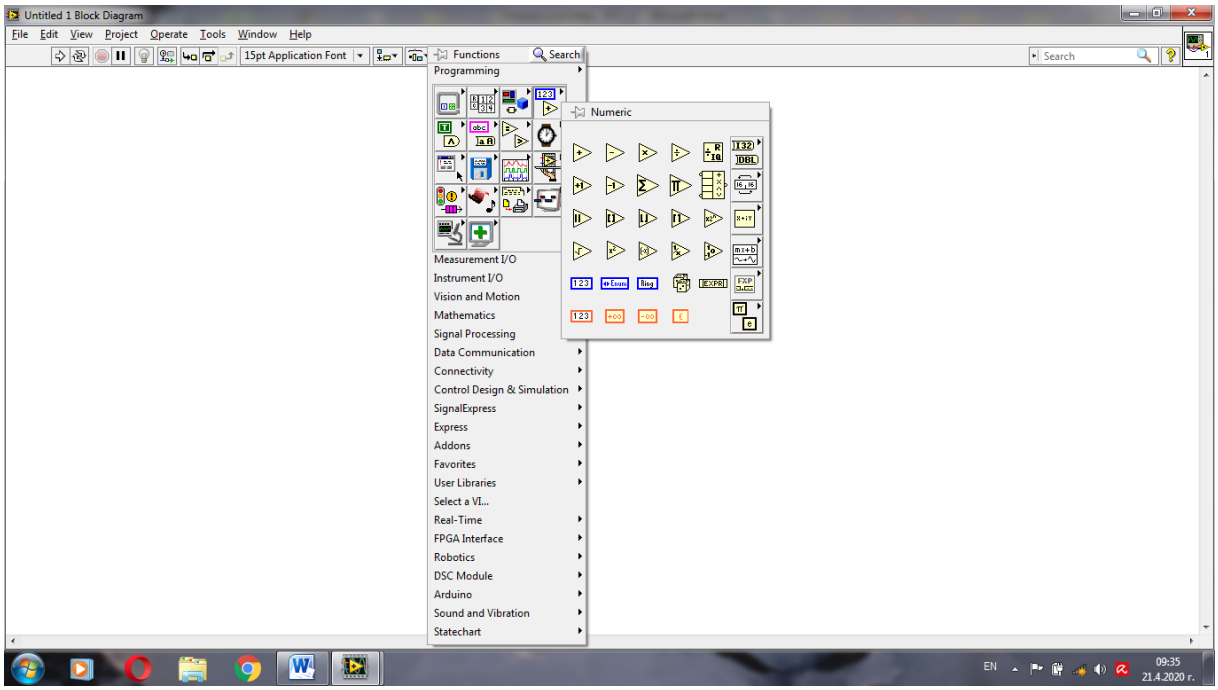


Фиг. 3.1. *Палети с инструменти в прозореца на компонента „Преден панел“*

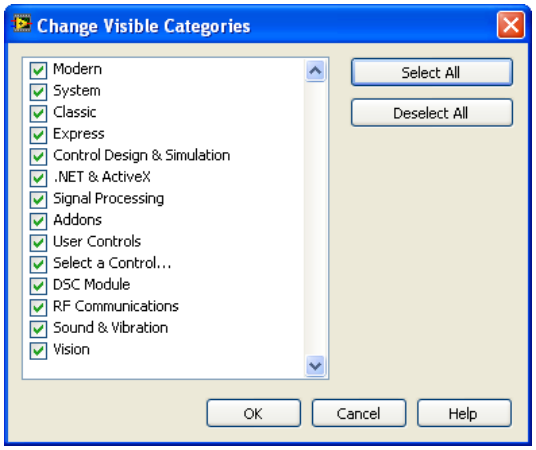
Предният панел е тази част от виртуалния инструмент, чрез която разработчиците имат възможността за управление на програмата:

- ✓ задаване на входни променливи;
- ✓ дефиниране и регулация на начални условия;
- ✓ фиксиране на мащабите при визуализация;
- ✓ прекратяване на програмното изпълнение.

Интерфейсът на предния панел (фиг. 3.1) включва менюта с индикатори и контроли, които за необходими за изграждане на приложенията – генератори на сигнали, бутони, вертикални и хоризонтални скали, дисплеи, графични панели и др. Блоковата диаграма (фиг. 3.2) е програмния компонент, където входовете и изходите на различните индикаторни и контролни елементи се свързват алгоритмично в модулни звена. Действията в блоковата диаграма се основават на парадигмата за „потоци от данни“. В нея се съдържат набори от библиотечни функции за реализация на различни типове обработки на измервателна и друга информация.



Фиг. 3.2. *Палети с библиотечни фунции в компонента „Блокова диаграма“*

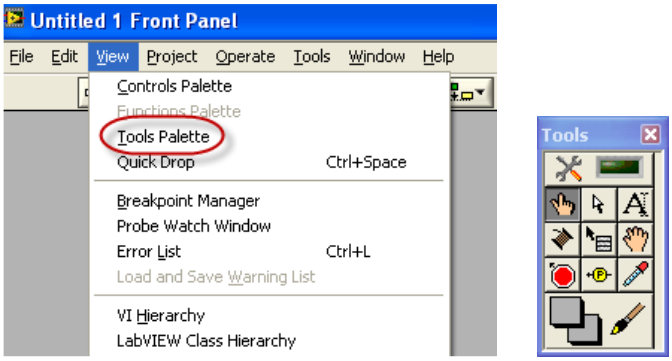


Фиг. 3.3. *Инструмент “Change Visible Categories”*

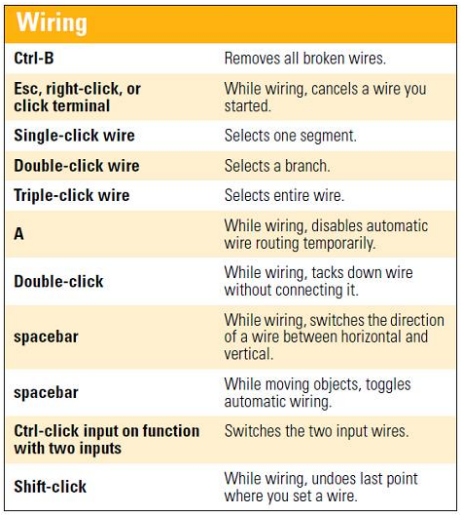
Посредством опцията “Change Visible Categories” се дава възможност за включване или промяната на конкретна библиотечна категория или група от функционални библиотеки, показано на фиг. 3.3.

Подменю „Tools” и инструмент “Wiring” на фиг. 3.4 и фиг. 3.5 са основни средства при създаване, изграждане на структурни връзки между контролните и индикаторните елементи и модифициране на виртуалните приложения. Комплексът от функции за стартиране на VIs (Virtual Instruments), форматиране и позициониране е даден на фиг. 3.. Описанието на компонентите на подменю „Tools” и инструмент “Wiring” е следното [6]:

-  *Operating tool* - служи за промяна стойностите на контрол. Когато мишката се намира над показалеца, курсора автоматично получава достъп до работния инструмент;
-  *Positioning tool* - използва се за позициониране, избор или промяна размера на обекти;
-  *Labeling tool* – предназначен е за поставяне на етикети и въвеждане на текст в контролен прибор;
-  *Wiring tool* – служи за свързване между контролите и индикаторите.



Фиг. 3.4. *Извикване на подменю “Tools”*



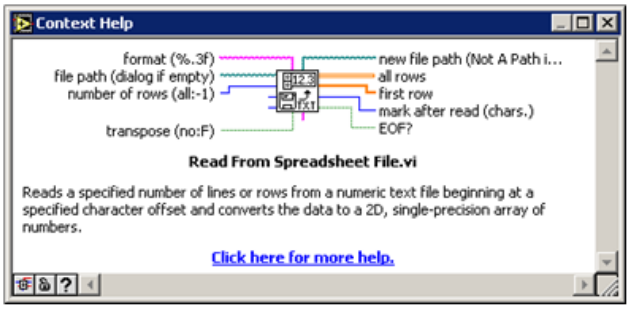
Wiring	
Ctrl-B	Removes all broken wires.
Esc, right-click, or click terminal	While wiring, cancels a wire you started.
Single-click wire	Selects one segment.
Double-click wire	Selects a branch.
Triple-click wire	Selects entire wire.
A	While wiring, disables automatic wire routing temporarily.
Double-click	While wiring, tacks down wire without connecting it.
spacebar	While wiring, switches the direction of a wire between horizontal and vertical.
spacebar	While moving objects, toggles automatic wiring.
Ctrl-click input on function with two inputs	Switches the two input wires.
Shift-click	While wiring, undoes last point where you set a wire.

Фиг. 3.5. *Инструмент “Wiring”*

Фиг. 3.6 *Лента с инструменти за изпълнение на VI, форматиране на текст и позициониране на обекти*

Във връзка с изпълнението на виртуалните приложения, както и възможностите за форматиране и позициониране на текстово съдържание и обекти, се използва специфицирана лента с инструменти, разполагаща с бутоните:

- *Run* – еднократно стартиране на VI;
- - изглед на бутон *Run* при стартиран виртуален инструмент, а не sub VI;
- - изглед на бутон *Run* при стартиран sub VI;
- - изглед на *Run* при наличие на грешки в приложението. Указва, че то не може да бъде стартирано;
- - циклично (непрекъснато) изпълнение на VI;
- - бутон *Abort Exsecution бутон*. Служи за преустановяване на изпълнението на инструментите;
- - *Pause* бутон, изпълнението на VI се поставя на пауза;
- - *Align Objets* бутон. Предназначен е за вертикално и хоризонтално подравняване/центриране на обекти;
- - чрез бутон *Distribute Objects* се осъществява разпръскване на групирани обекти в различни направления;
- *Resize Objects* бутон. Служи за преоразмеряване на индикаторни и контролни елементи;
- - бутон *Reorder* е предназначен за изместване или поставяне на даден обект пред или зад позициониран друг обект.



Фиг. 3.7. *Show Context Help*

От меню "HELP" може да бъде изискана основна информация за предназначението, функционалните входове и изходи на обектите в LabVIEW посредством подменю "Show Context Help", показан на фиг. 3.7. За достъп до подменюто курсорът на мишката трябва да се постави върху конкретен желан обект по избор. При негово придвижване върху обектите от предния панел или блоковата диаграма, "Context Help Window" извежда подробно структурно описание [5, 6].

Мрежови протоколи Transmission Control Protocol (TCP) и User Datagram Protocol (UDP)

TCP протоколът поддържа надеждност при предаване на информационни данни или „сегменти" между комуникиращи мрежови хостове с известно внесено забавяне на информационния трансфер, с проследяване на големината на потока от сегменти, скоростта на предаване и натоварването на мрежовия трафик. Всеки изпратен байт информация разполага с назначен номер, като при получаване от страна на приемащия

хост се изпраща съответно потвърждение към източника. При достигане до хоста получател следва подреждане на изпратените сегменти според назначения пореден номер, така че да се получи коректното оригинално съобщение. Всеки TCP сегмент разполага с 20 байта служебна информация. Цялостната процедура, свързана с гарантиране на сигурността при доставка на пакетни данни, преминава през следната последователност:

- Установяване на унифицирана сесия между двата комуникиращи хоста;
- Контрол на последователността на данните в предаваните пакети;
- Контрол на трафика от данни с управление на капацитета за буфериране;
- Поддържане на връзка при липса на данни за предаване между съответните хостове;
- Генериране на потвърждение за коректност на приети данни;
- Затваряне на сесията между комуникиращите хостове.

TCP е подходящ за приложения като WEB браузъри и сървъри, E-mail и файлов трансфер (FTP), но не е препоръчителен за приложения в реално време като Voice over IP и Video over IP.

Протоколът UDP гарантира високо бързодействие при предаването на единици данни или „дейтаграми“, без проследяване на поредността и проверка на коректността на тяхното получаване при достигане до крайното мрежово местоназначение. Стандартът се грижи единствено за цялостта на данните чрез контролна сума. Това предполага неговото по-ниско ниво на надеждност спрямо Transmission Control Protocol. Дължината на дадена UDP дейтаграма може да достигне до 65515 байта, разполагайки само с 8 байта служебна информация. Големите дайтаграми следва да бъдат фрагментирани, но тяхната употреба не е препоръчителна. Програмите, базирани на протокола, е необходимо да разполагат със следните възможности:

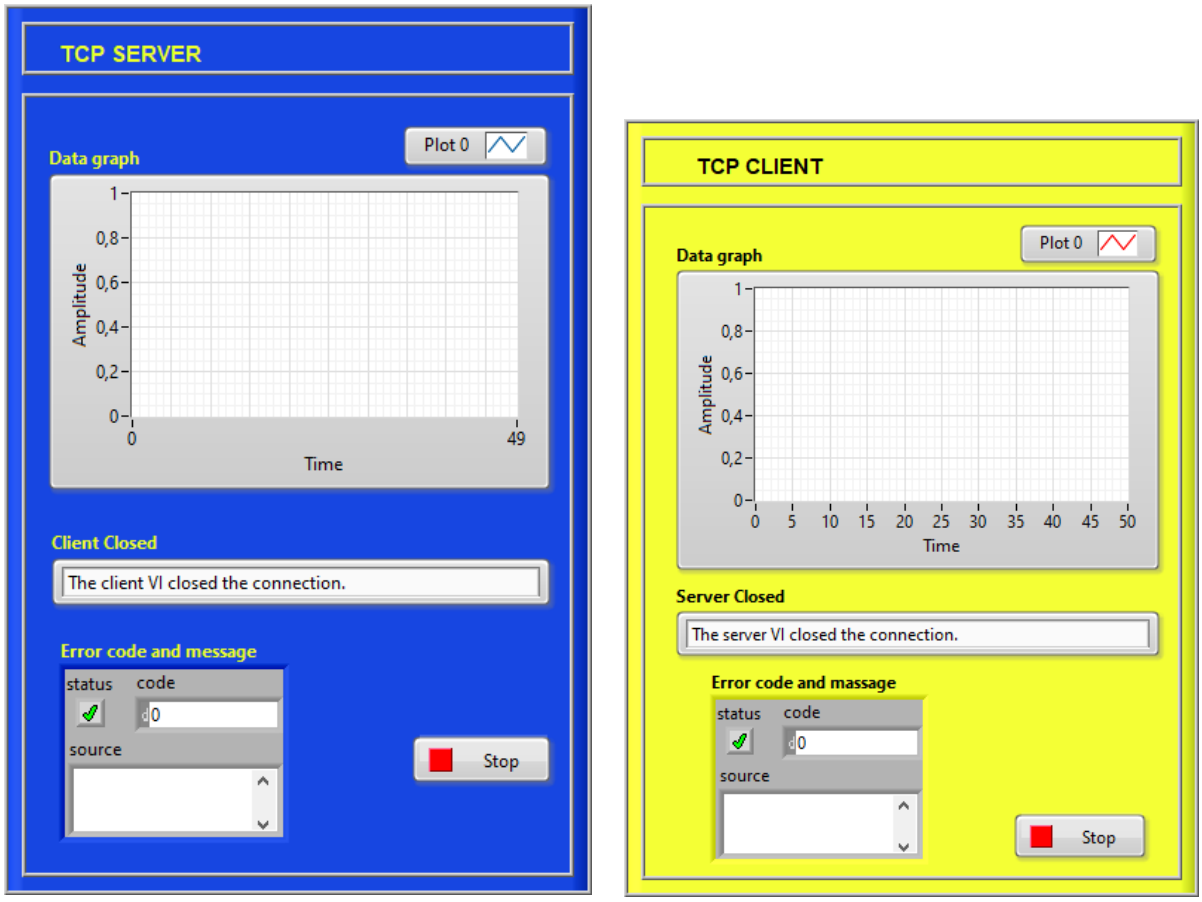
- Препредаване на загубени единици с данни;
- Игнориране на повторения;
- Фрагментиране и обединяване на големи информационни потоци в обхвата на дейтаграмите.

Основните приложения на UDP протокола се свързват с VoIP, Internet Radio, видео и аудио в реално време, което е предимство в сравнение с TCP. Тук също могат да включени Domain Name Service (DSN), мрежови мониторинг и управление, предаване на двоични файлове или Trivial FTP. Поддържа услугите Multicast (изпращане на пакетни данни до зададена група/категория потребители) и Broadcast (пакетно предаване на данни до всички мрежови устройства в структурата на обособената локална или част от глобалната мрежа) [5, 6].

3.3. Последователност на работа и задачи за изпълнение

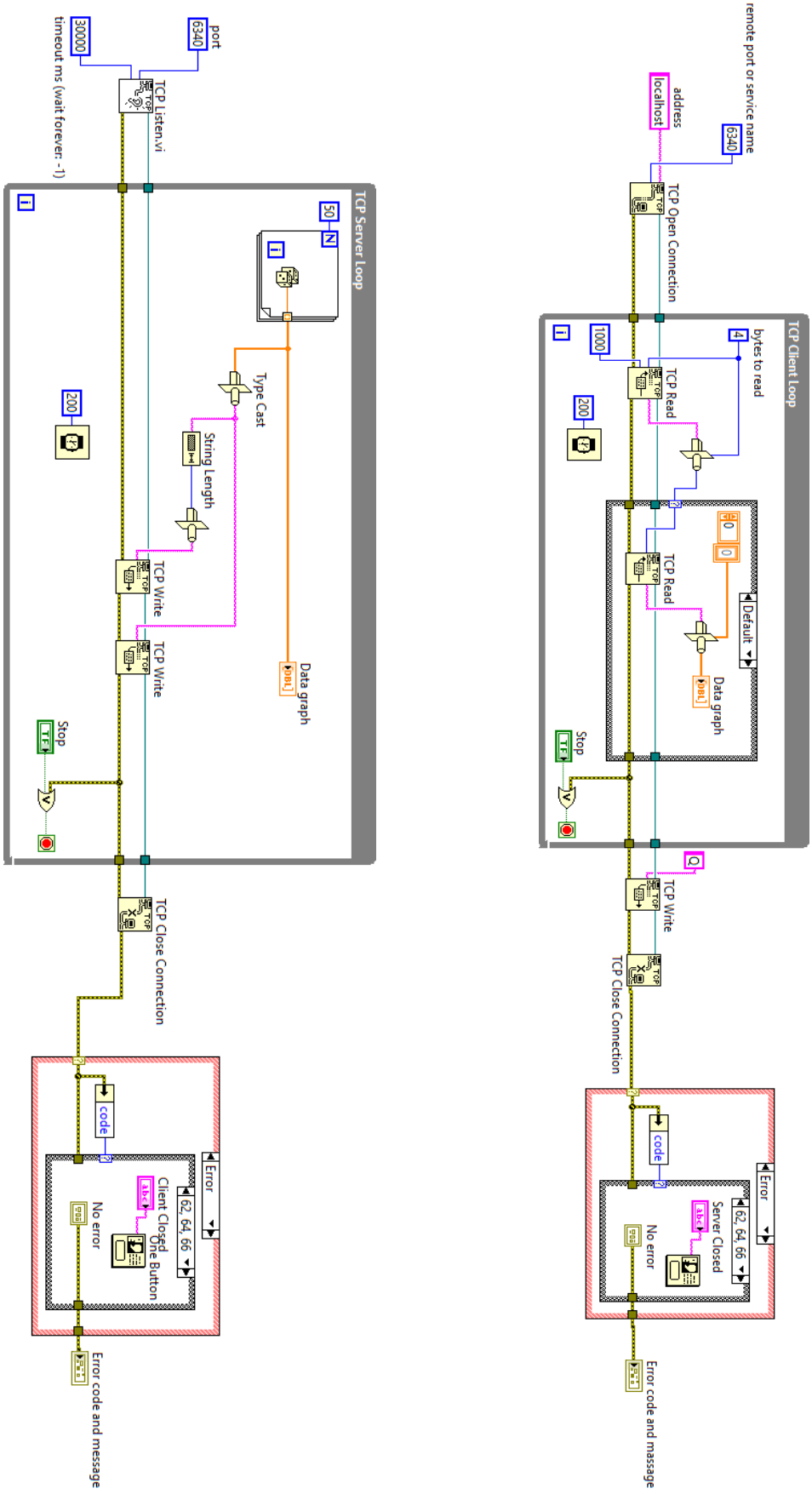
- Моделиране на виртуални инструменти, изпълняващи ролята на системни мрежови звена „сървър“ и „клиент“, за обмен на информационни пакети чрез библиотечни функции за интеграция на Transmission Control Protocol, следвайки потребителските интерфейсни и блоковите диаграми от фиг. 3.8 и 3.9. *При изпълнение на задачата да бъде следвано функционалното описание:*

„Върху панелите се разполагат графични индикатори за визуализация на изпращаните и получаваните данни от информационен източник „генератор на случайни числа“, индикатори за код и дефиниция на грешки относно отделните процедури. Създаването и затварянето на унифицирана TCP връзка, предаването и получаването на пакетни данни в символен формат се реализират на базата на библиотечните функции “TCP Open Connection”, “TCP Close Connection”, “TCP Write” и “TCP Read”. Указаните последни две библиотечни функции се използват два пъти в структурата на всяка от диаграмите с цел определяне на количеството предавани/приемани данни и същинските процеси на техния запис и прочитане. Генерирането на TCP информационни сегменти се осъществява чрез програмен фрагмент с циклична организация с включване на „генератор на случайни числа“. Данните се предават между сървърното и клиентското приложение по съответен зададен source (приложение, иницииращо сесията) и присвоен destination (приложение-дестинация, указващо работещ отдалечен мрежови хост) порт с номер 6340. При двата инструмента се фиксира еднакво времезакъснение във връзката с изработването, предаването и получаването на пакетни данни. В края на структурата на блоковата диаграма на клиенсткото приложение се използва относно функционалният елемент “TCP Write”, чието предназначение се отнася до изпращането на единичен символ “Q” (индикация към сървъра за прекратена активност на връзката от страна на клиента).;



а) *б)*

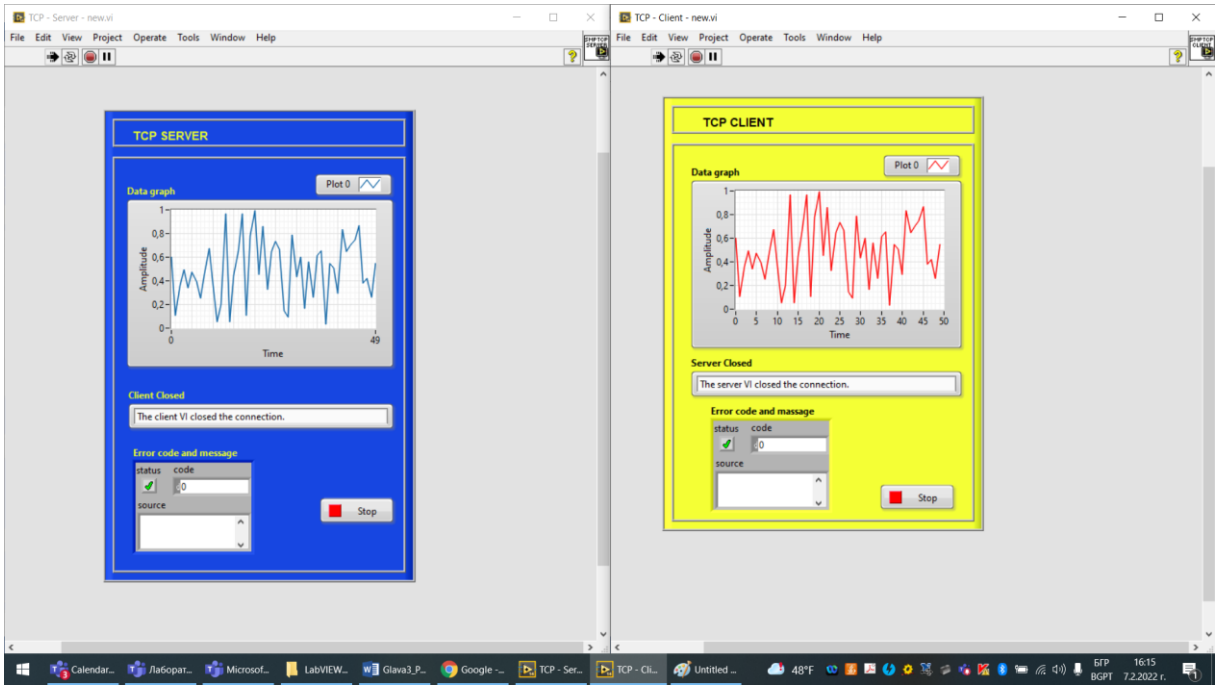
Фиг. 3.8. *Предни панели на TCP виртуални приложения а) „сървър“ и б) „клиент“*



а) *б)*

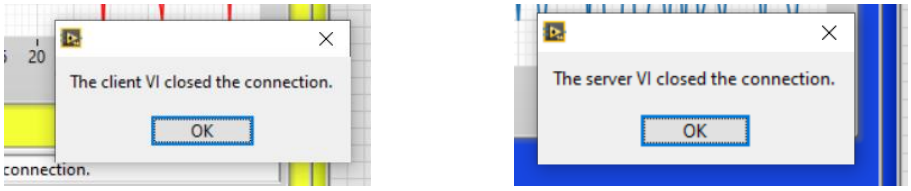
Фиг. 3.9. *Блокови диаграми на TCP виртуалните приложения а) „сървър“ и б) „клиент“*

❖ Тестване на работоспособността на потребителските TCP виртуални приложения чрез стартиране на тяхното циклично изпълнение в паралел. При коректност на процедурата следва да бъде наблюдаван идентичен изглед при сегментното предаване върху предвидените графични и цифрови индикатори, както е показано на фиг. 3.10;



Фиг. 3.10. *Изглед на информационния трансфер на функционални TCP виртуални приложения а) „сървър“ и б) „клиент“*

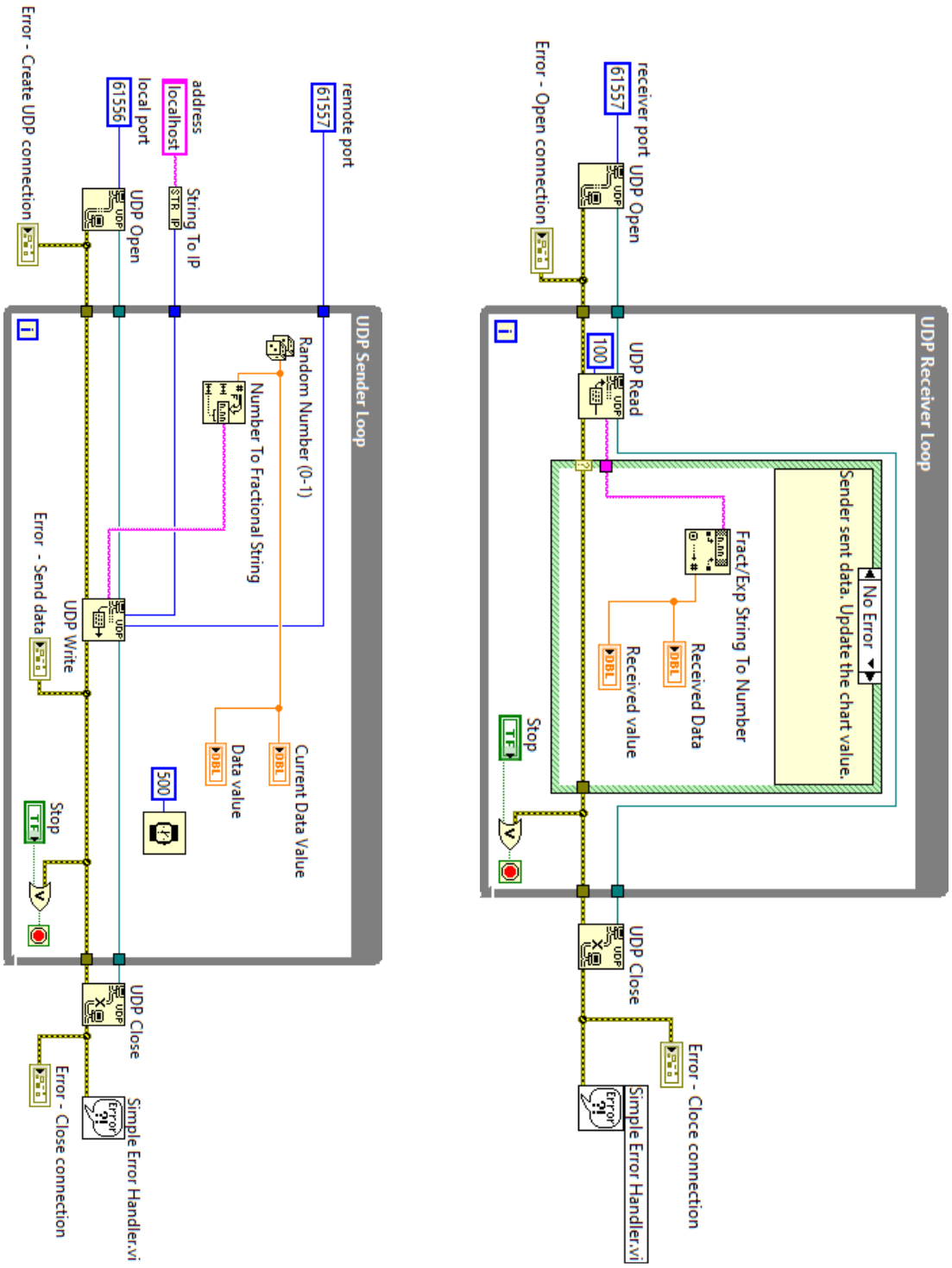
❖ Проверка на коректността на извеждане на предвидени диалогови прозорци и прекратяване на изпълнението на инструментите при пресутановяване на връзката/сесията от страна на сървърното и клиентското виртуално приложение. Процесът е демонстриран на фиг. 3.11;



а) *б)*

Фиг. 3.11. *Диалогови прозорци при преустановяване на връзката от страна на а) клиентското и б) сървърното приложение*

❖ Създаване, конфигуриране и тестване на виртуалните приложения за предаване и получаване на UDP дейтаграми “Sender” и “Receiver”, базирайки се на системните блокови диаграми и потребителските интерфейсни панели, представени на фиг. 3.12 и фиг. 3.13. По-долу е дадено кратко пояснително описание относно функционалността на инструментите:

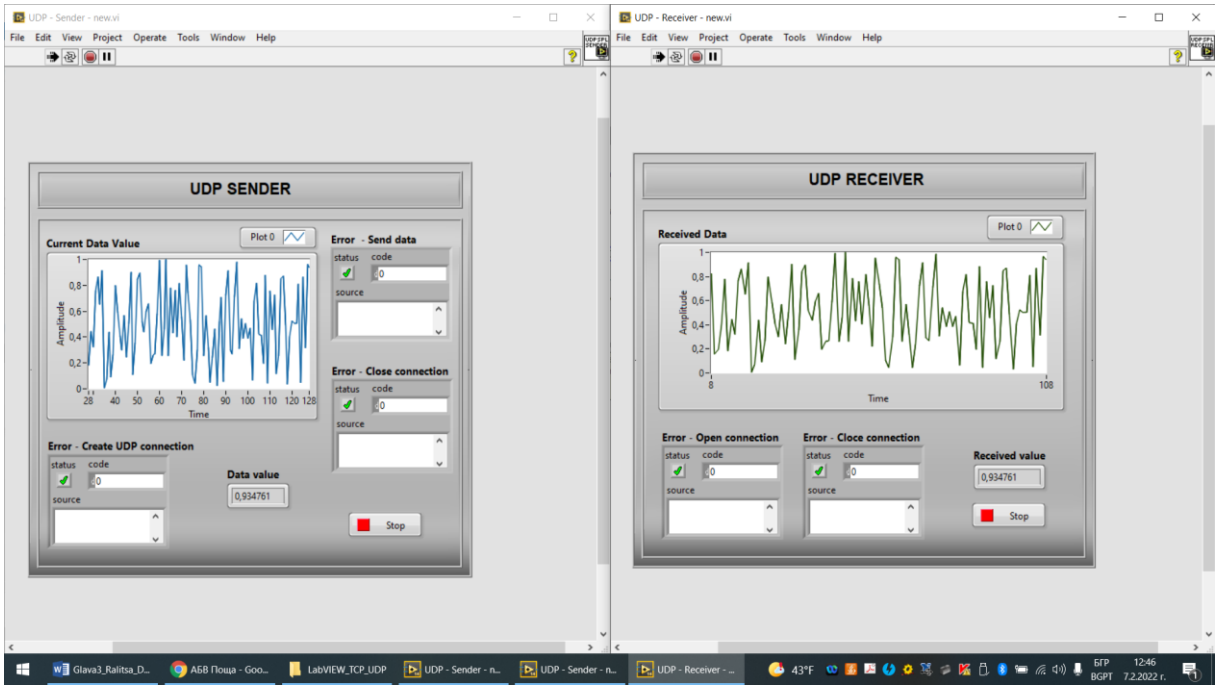


а) *б)*

Фиг. 3.12. *Функционални блокови диаграми на VI а) „предавател“ и б) „приемник“*

В случая в ролята на източник на пакетни съобщения относно се използва „генератор на случайни числа“. Преди данните да бъдат изпратени до дефиниран отдалечен порт 65557 е необходимо да бъдат конвертирани от double в символен тип. Освен отдалеченият порт за предаване на данни трябва да бъде оказан и клиентът, на който ще е стартиран процесът на предаване - това се извършва чрез символна константа, която се преобразува в актуалния IP адрес. След като информационните масиви бъдат получени се извършва обратното преобразуване

от символен в числов тип с двойна прецизност и непосредствена визуализация. Панелите на инструментите разполагат с графични и числови индикатори за наблюдение на предаваните и получаваните пакетни данни. Индикатори за грешки ще изведат описание и кода на грешката по отношение на създаване и затваряне на UTP връзка, изпращане на данни“;



Фиг. 3.13. *Предни панели на VI „предавател“ и „приемник“ при трансфер на съобщения в реално време*

❖ Проверка на състоянието при изпращане на съобщения от програмното звено UDP Sender при ситуация на прекратена връзка от страна на моделирания в LabVIEW „мрежови приемник“.

3.4. Контролни въпроси

1. Кои са модулните компоненти и функционални елементи, изграждащи виртуалните инструменти в графичната среда LabVIEW?
2. През каква последователност протича изпрашането на информационните сегменти при мрежови протокол Transmission Control Protocol?
3. Кои са предимствата и недостатъците на мрежовия протокол TCP?
4. Какво отношение има използването на „контролна сума“ към получаваните дейтаграми при UDP протокола?
5. В кои приложения User Datagram Protocol е предпочитан спрямо TCP?

Лабораторно упражнение № 4

СЕГМЕНТИРАНЕ И УПРАВЛЕНИЕ НА ТРАФИКА В ЖИЧНИ МРЕЖОВИ КОНФИГУРАЦИИ В СРЕДА CISCO PACKET TRACER

4.1. Цел на упражнението

Да се запознаят студентите с програмните възможности и инструменти на симулатор Cisco Packet Tracer за моделиране, конфигуриране и управление на пакетното предаване на данни при конфигурации с мрежово сегментиране.

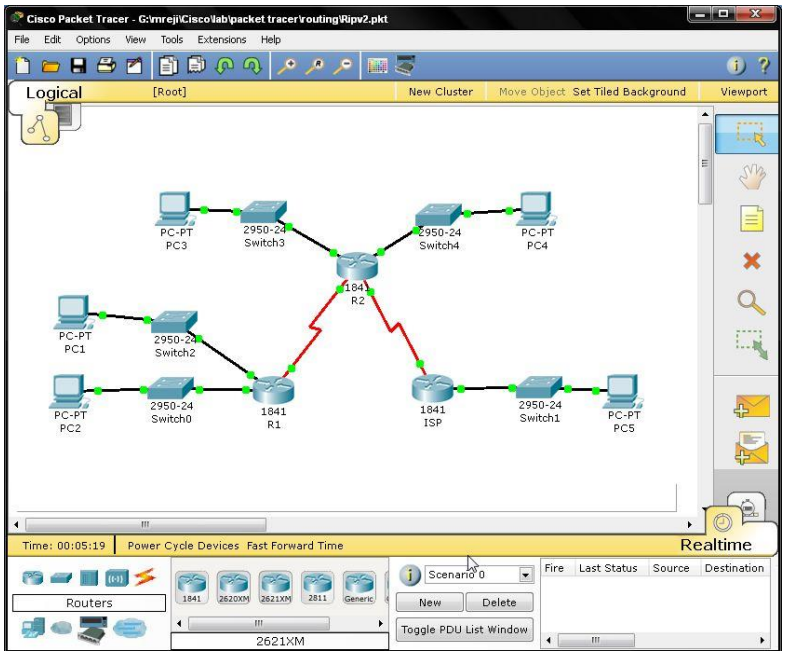
4.2. Теоретична постановка

Мрежови симулатор Cisco Packet Tracer

Софтуерът Cisco Packet Tracer е мощен мрежов симулатор, който може да се използва при обучение за мрежово сертифициране, като позволява на обучаемите да създават мрежи с почти неограничен брой устройства и да извършват отстраняване на неизправности, без да се налага купуването на съответния хардуер на Cisco. Инструментът е създаден от Cisco Systems. Целта на Packet Tracer е да предложи на потребителите инструмент за усвояване на принципите на работа в мрежа. Проследяването на пакетите позволява създаването на мрежа чрез влачене и пускане на мрежовите компоненти и свързване към определен порт на дадено устройство. Свързването на група компютри и задаването на техни IP адреси в обхвата на една на съща мрежа обособява мрежова инфраструктура, която се нарича Local Area Network (LAN) [7].

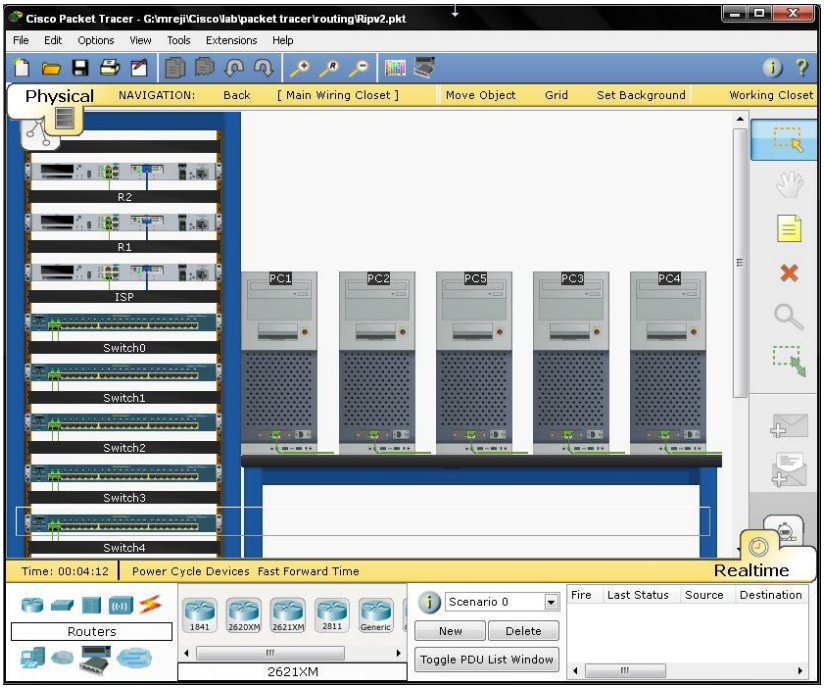
Според някои източници Packet Tracer може да бъде разгледан като симулатор на протоколи, разработен от Денис Фрецо и неговия екип в Cisco Systems. Средата е мощен и динамичен инструмент, който показва различните протоколи, използвани в мрежата, или в реално време, да бъдат илюстрирани в режим на симулация. Това могат да бъдат:

- протоколи от ниво 2 като Ethernet и PPP (Point-to-Point);
- протоколи от ниво 3 като IP (Internet Protocol), ICMP (Internet Control Message Protocol) и ARP (Address Resolution);
- протоколи от ниво 4 като TCP и UDP.

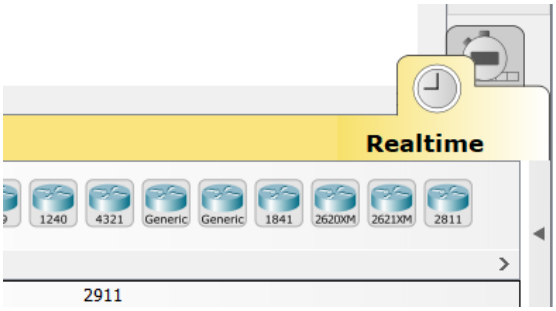


Фиг. 4.1. *Логическа работна среда*

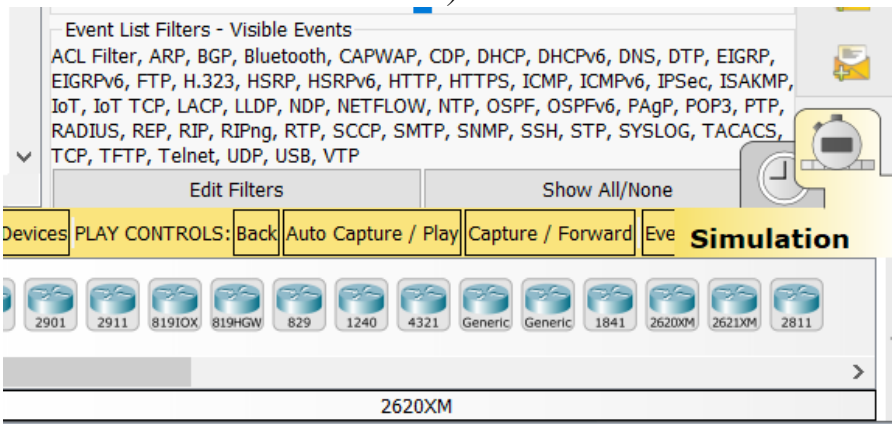
Packet Tracer има два вида работни среди – *логическа* и *физическа*. *Логическата среда*, представена на фиг. 4.1, позволява на потребителите да изграждат логически мрежови топологии като добавят, свързват и групират мрежови устройства. *Физическата среда*, показана на фиг. 4.2, изобразява графично логическата мрежа и представя мрежовите устройства (рутери, суичове и т.н) как биха изглеждали реално.



Фиг. 4.2. *Физическа работна среда*



а)



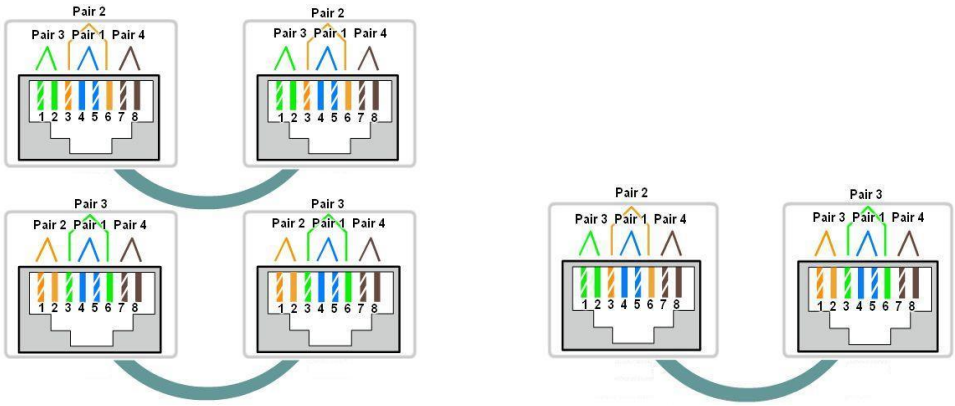
б)

Фиг. 4.3. *Преминаване от режим а) „Realtime“ към б) “Simulation”*

Има два вида работни режими в Packet Tracer – *режим на реално време* и *симулационен режим*. В режим на *реално време* поведението на мрежата е като с реални устройства, с незабавни отговори в реално време за всички действия на мрежата. Реалният режим дава алтернативен вариант на реалното оборудване и осигурява предварително практическо обучение. В *симулационния* режим потребителят може да виджда и управлява времевите интервали, трансфера на данни и тяхното разпространение в мрежата. Това спомага да се разберат основните концепции на мрежовите операции. Преминаването от единия към другия режим е показано на фиг. 4.3 [8].

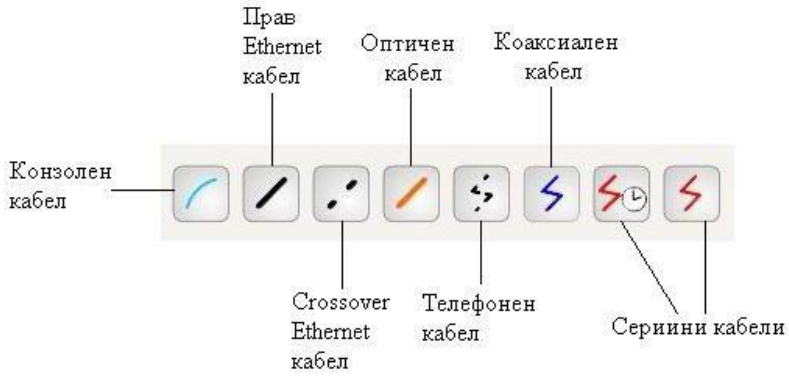
Изисквания за свързване между мрежовите устройства.

Ако два RJ-45 конектора на кабел се поставят с една и съща страна, така че да се виждат цветовите кодировки на проводниците и подредбата на всеки от тях съответства на номера, който заема в конектора, тогава е налице *право свързване* или *straight-trought*. При *обратно свързване* или *Crossover*, RJ-45 конекторите на двата края показват, че проводниците от единия край са кръстосани и влизат в различни пинове, както е показано на фиг. 4.4. На фиг. 4.5 е дадено описание на някои видове свързвания в Cisco Packet Tracer.



а) *б)*

Фиг. 4.4. *Straight-throught а) и Crossover б) тип на свързване*



Фиг. 4.5. *Видове свързвания в Cisco Packet Tracer*

.

Straight-trought тип се използва в следните случаи на свързване:

- ❖ Суич към рутер;
- ❖ Суич към компютър или сървър;
- ❖ Хъб към компютър или сървър.

Crossover тип се използва при следните случаи на свързване:

- ❖ Суич към суич;
- ❖ Суич към хъб;
- ❖ Хъб към хъб;
- ❖ Рутер към рутер;
- ❖ Компютър към компютър;
- ❖ Рутер към компютър [8].

4.3. Последователност на работа и задачи за изпълнение

Сегментиране на мрежова конфигурация в три зони на трафично потребление

Процесът обособяване на подмрежови зони в рамките на мрежа с корпоративно, административно или друго предназначение се отнася като възможност за изграждане на оптимални мрежови архитектури. Посредством тези процедури може да се постигне максимално редуциране на заявки от страна на:

- ✓ зловреден софтуер;
- ✓ неоторизиран потребителски достъп;
- ✓ други видове атаки.

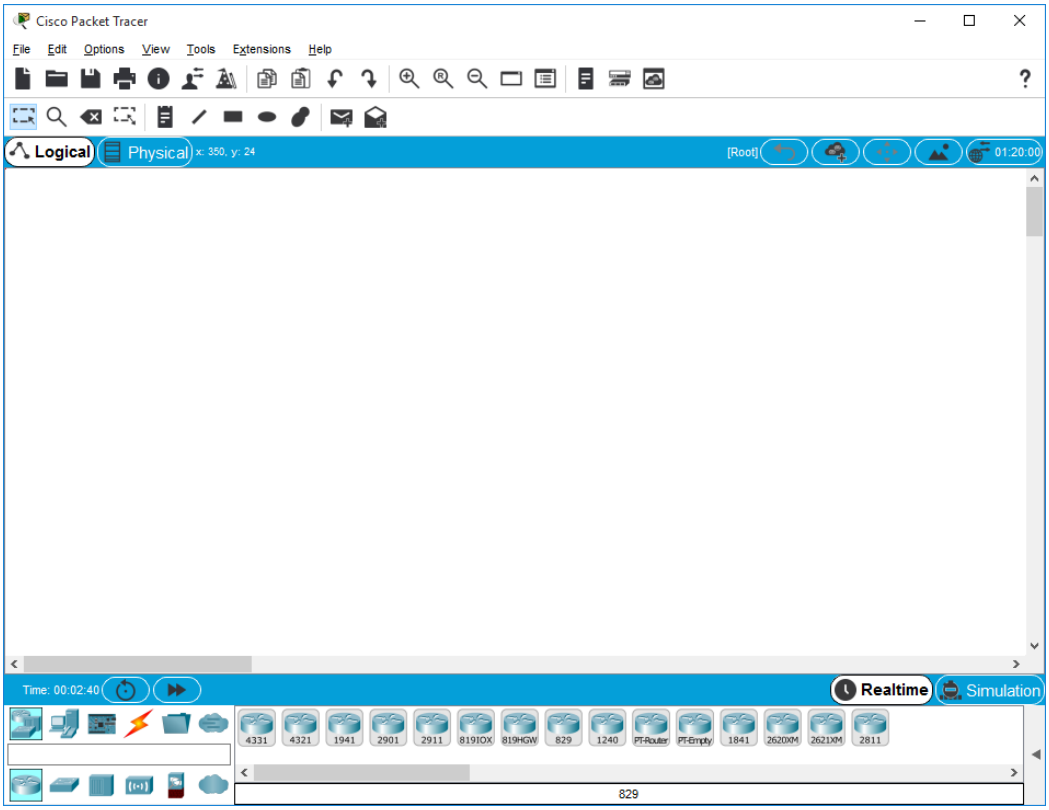
Мрежовото сегментиране и конфигуриране на пакетното предаване на данни може да бъде предварително извършено в условията на симулационна среда. След процеси на изследване на коректността и бързодействието на предавания трафик между структурните мрежови устройства, може да бъде преминато към реално изграждане на обектната мрежова инфраструктура в съответен офис или друг административен център при гарантиране на QoS на услугата.

За целта може да бъде използван симулаторът Cisco Packet Tracer като по-долу е дефинирана стъпковата последотелност във връзка с изграждането, конфигурацията и тестването на информационния трансфер на данни в обхвата на мрежовите устройства с динамично назначаване на IP адресите в рамките на три подмежови трасета, както следва:

- ✓ Подмрежа Net №1: 192.168.10.0 to 192.168.10.63;
- ✓ Подмрежа Net №2: 192.168.10.64 to 192.168.10.127;
- ✓ Подмрежа Net №3: 192.168.10.128 to 192.168.10.191.

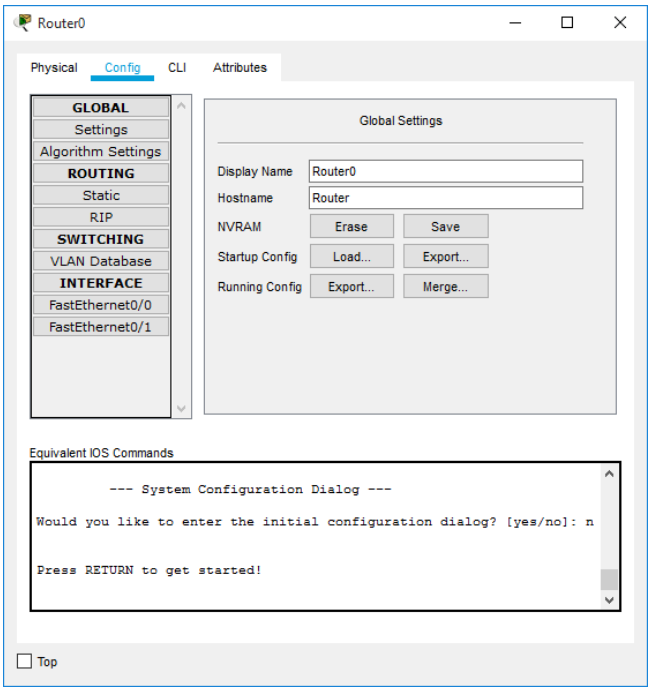
а)

Фигура 4.6 онагледява процеса на зареждане и началния изглед на средата Cisco Packet Tracer - версия „7.3“. Във връзка с изграждане на мрежата могат да бъдат използвани мрежови устройства „маршрутизатор“ и „комутатори“ модели „2811“ и „2950-24“. Избраният модел Router 2811 разполага с два налични Fast Ethernet интерфейса – fa 0/0 и fa 0/1, дадено на фиг. 4.7.а).

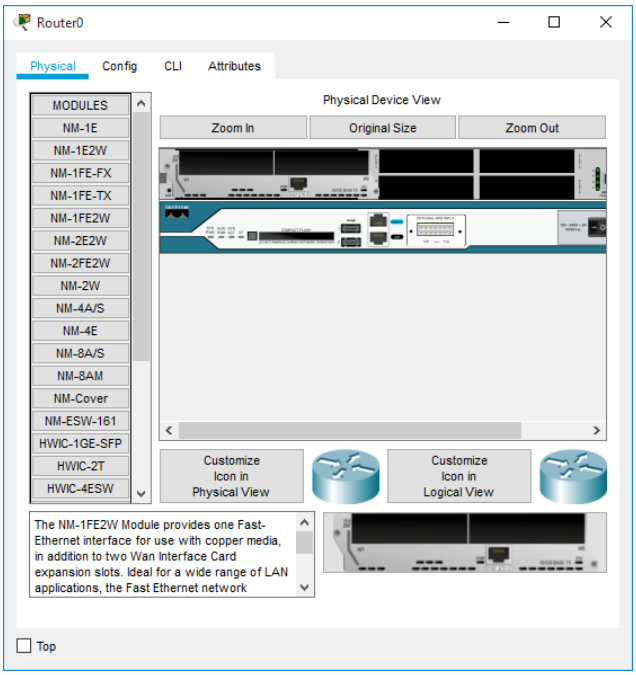


б)

Фиг. 4.6. *Ход на зареждане а) и начален изглед б) на симулатор Cisco Packet Tracer*



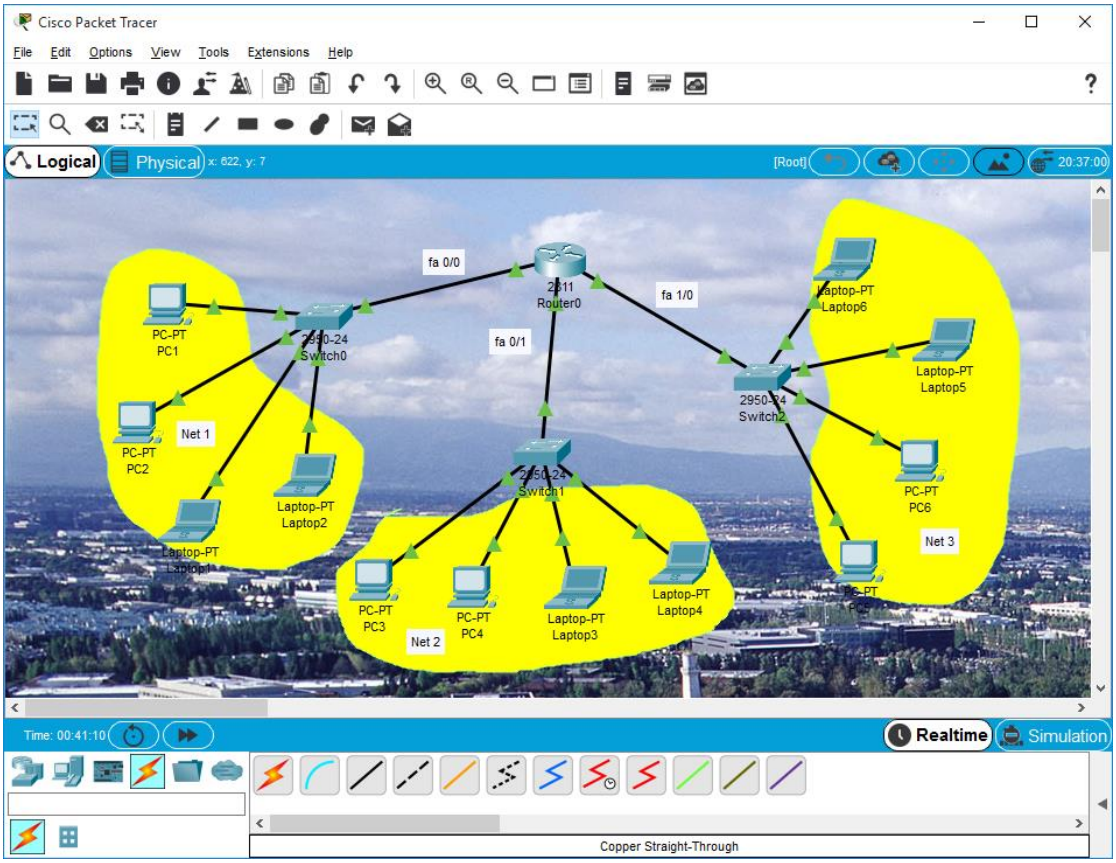
а)



б)

Фиг. 4.7. *Налични fa портове а) и въвеждане на модул NM-1FE2W с включване на допълнителни fa интерфейси б) към рутер 2811*

Тъй като задачата изисква формиране на три подмрежови зони, е необходимо да бъде добавено интерфейсно звено с включване на поне един fa комуникационен порт. В случая е добавен модул NM-1FE2W - фиг. 4.7.б, чрез който се включва порт fa 1/0.



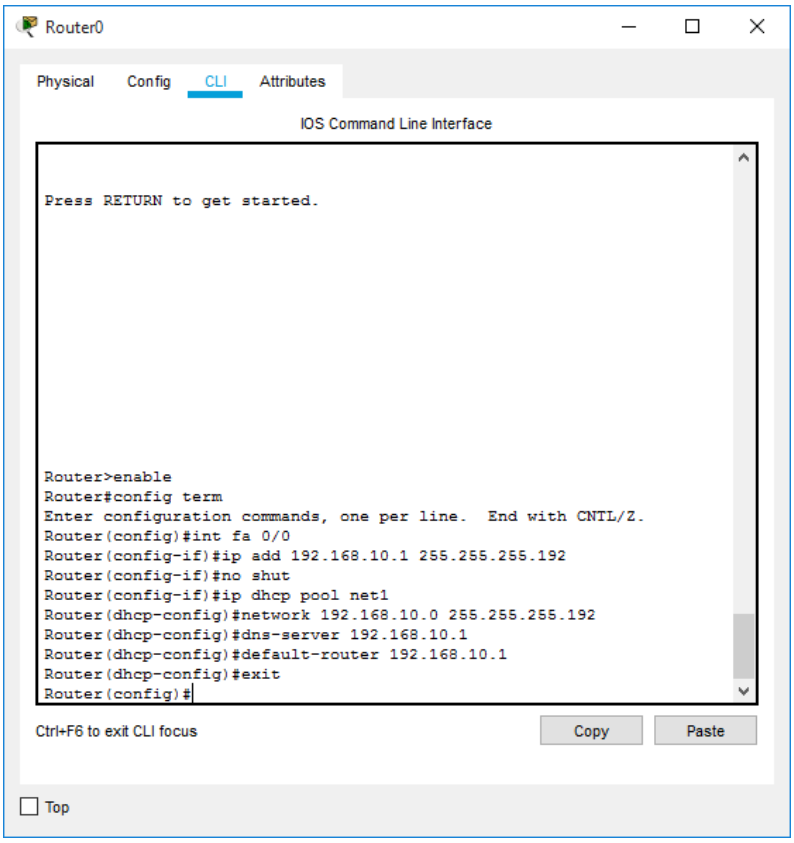
Фиг. 4.8. *Мрежова архитектура с четири подмрежови зони*

След добавяне на модула и осигуряване на необходимия брой интерфесни звена се преминава към построяване на мрежовата концепция. В структурата на всеки отделен сегмент могат да бъдат свързани до 25 персонални потребителски станции чрез прилагане на Copper Straight-Through тип на свързване. На фиг. 4.8 е даден крайният вид на мрежовата архитектура при обособени подмрежови зони.

Следващата стъпка се свежда до въвеждане на функционални набори по отношение на специфицирани конфигурационни параметри за всяка от обособените мрежови зони. Настройките се свързват с:

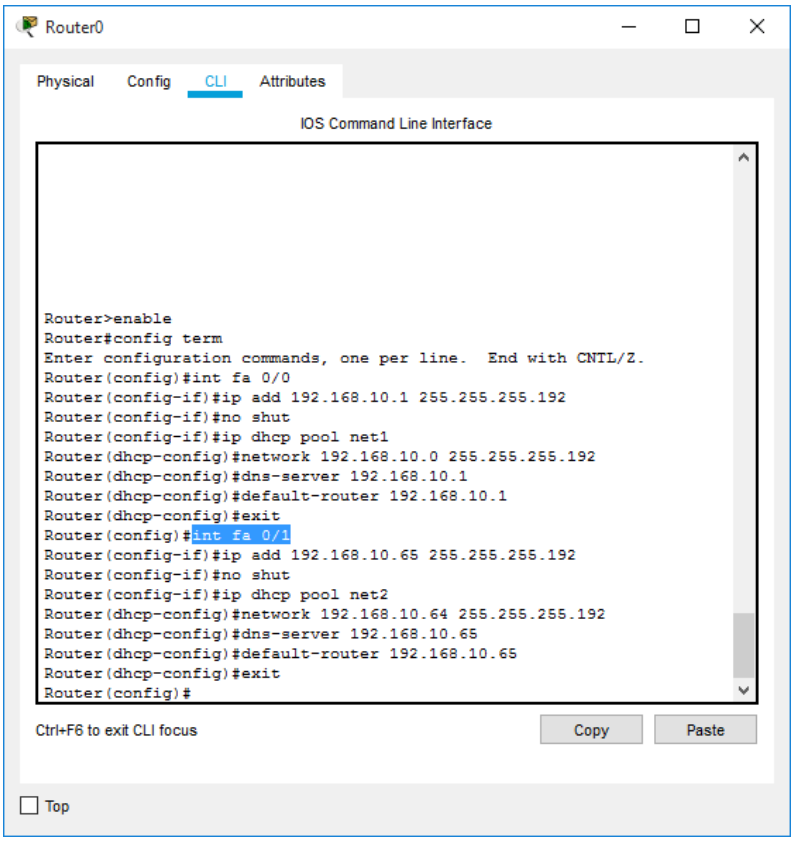
- ✓ задаване на IP адреси и подмрежови маски относно маршрутизация – интерфейси fa 0/0, fa 0/1 и fa 1/0;
- ✓ задаване на етикети/идентификационни номера на мрежовите сегменти и обхват на динамично назначавани IP адреси;
- ✓ DNS сървър и рутер по подразбиране.

Всички настройки се извършват в IOS Command Line Interface (CLI) на маршрутизиращото звено – Router 2811. Процесите са последователно представени от фиг. 4.9 до фиг. 4.11.

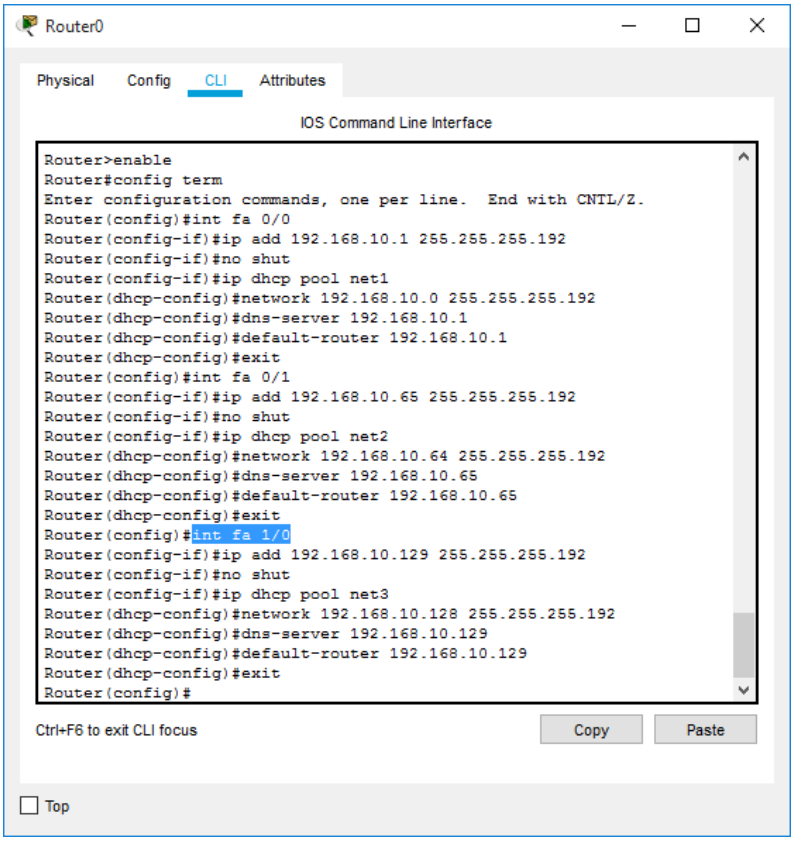


Фиг. 4.9. *Конфигурационни настройки на рутера за сегмент Net №1*

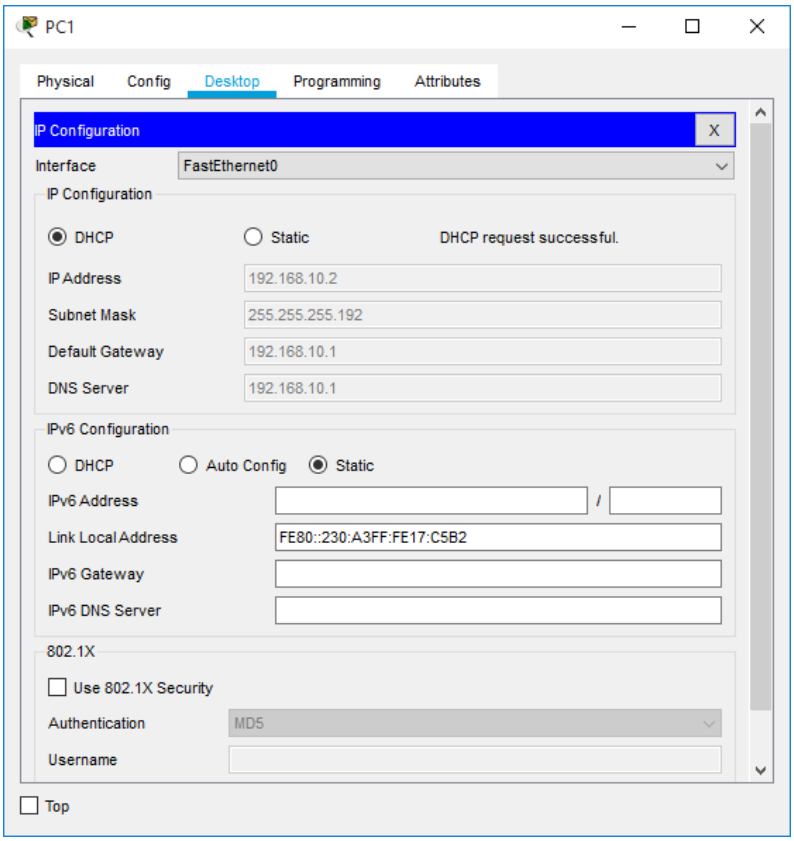
След въвеждане на изискваните конфигурационни параметри, касаещи обособените мрежови сегменти, следва спрямо всяка персонална компютърна станция в техния обхват да се реализира динамично назначаване на IP адреси. Процесът се извършва чрез избор на компютърна станция – Desctop -> IP Configuration на фиг. 4.12.



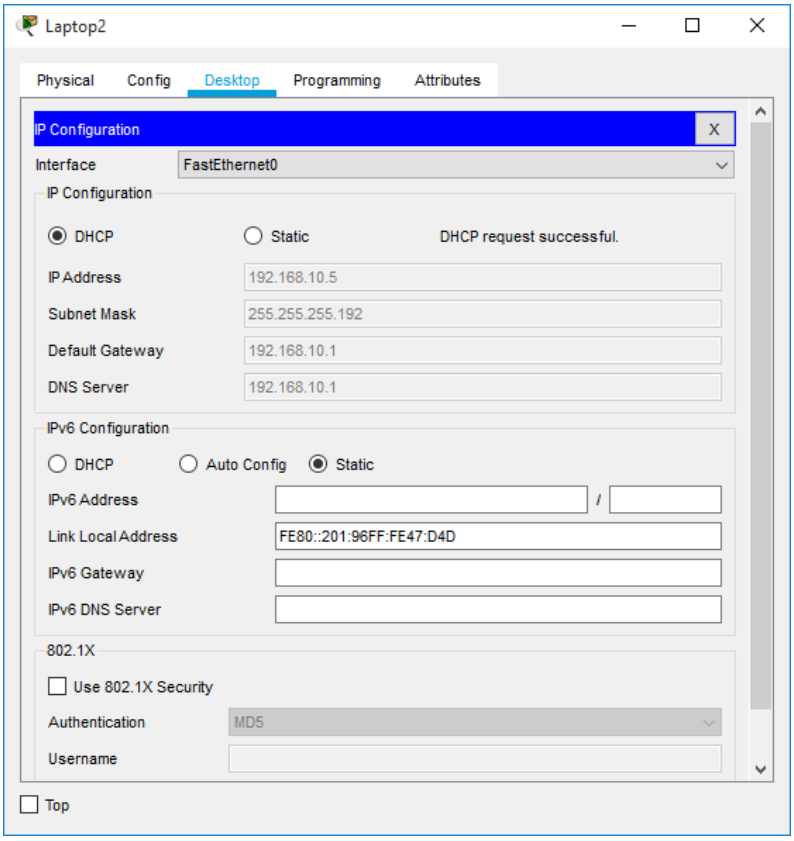
Фиг. 4.10. *Конфигурационни настройки на рутера за сегмент Net №2*



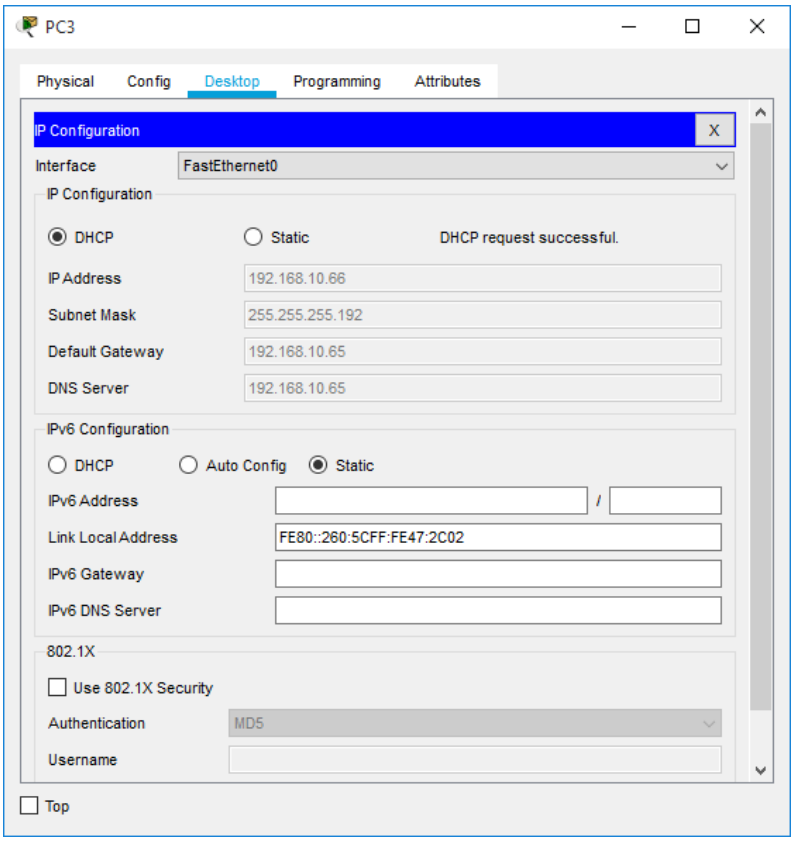
Фиг. 4.11. *Конфигурационни настройки на рутера за сегмент Net №3*


PC1
Physical
Config
Desktop
Programming
Attributes
IP Configuration
X
Interface
FastEthernet0
IP Configuration
DHCP
Static
DHCP request successful.
IP Address
192.168.10.2
Subnet Mask
255.255.255.192
Default Gateway
192.168.10.1
DNS Server
192.168.10.1
IPv6 Configuration
DHCP
Auto Config
Static
IPv6 Address
Link Local Address
FE80::230:A3FF:FE17:C5B2
IPv6 Gateway
IPv6 DNS Server
802.1X
Use 802.1X Security
Authentication
MD5
Username
Top

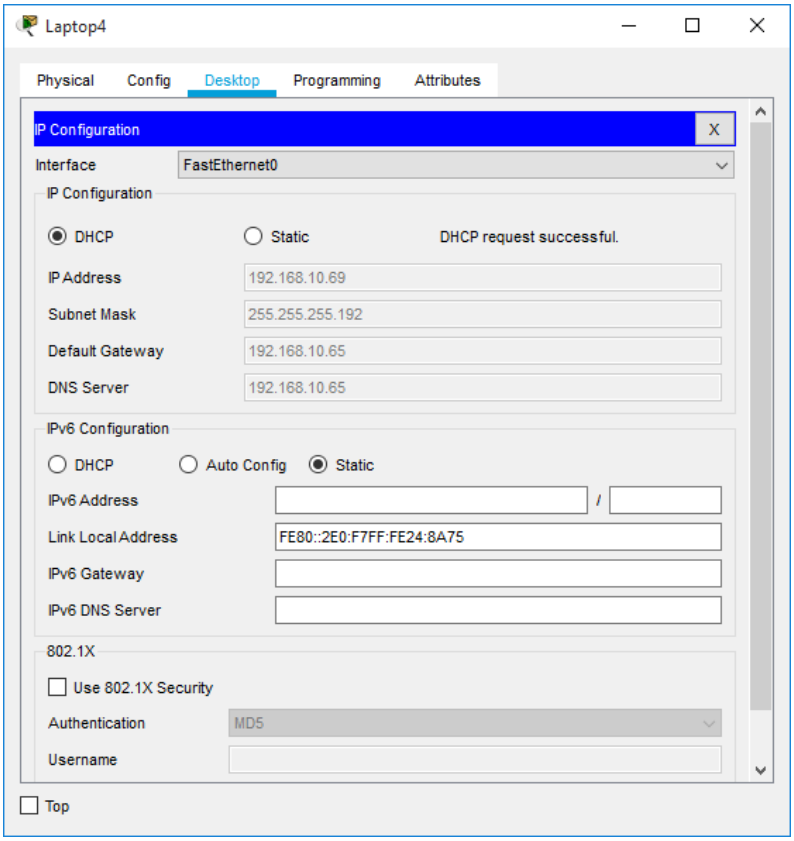
а)


Laptop2
Physical
Config
Desktop
Programming
Attributes
IP Configuration
X
Interface
FastEthernet0
IP Configuration
DHCP
Static
DHCP request successful.
IP Address
192.168.10.5
Subnet Mask
255.255.255.192
Default Gateway
192.168.10.1
DNS Server
192.168.10.1
IPv6 Configuration
DHCP
Auto Config
Static
IPv6 Address
Link Local Address
FE80::201:96FF:FE47:D4D
IPv6 Gateway
IPv6 DNS Server
802.1X
Use 802.1X Security
Authentication
MD5
Username
Top

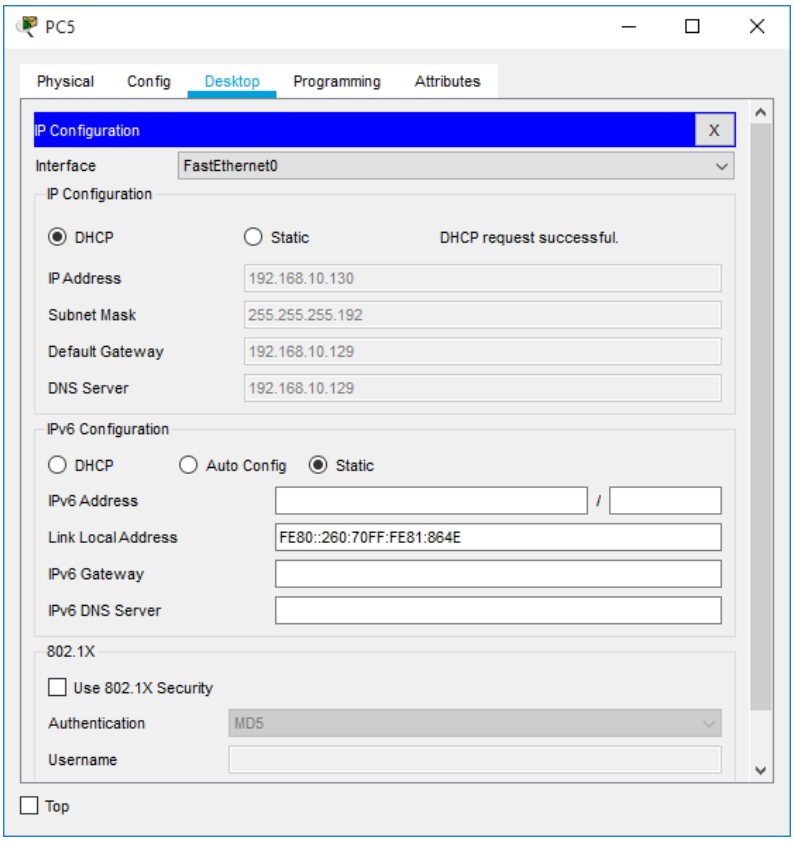
б)


PC3
Physical
Config
Desktop
Programming
Attributes
IP Configuration
X
Interface
FastEthernet0
IP Configuration
DHCP
Static
DHCP request successful.
IP Address
192.168.10.66
Subnet Mask
255.255.255.192
Default Gateway
192.168.10.65
DNS Server
192.168.10.65
IPv6 Configuration
DHCP
Auto Config
Static
IPv6 Address
Link Local Address
FE80::260:5CFF:FE47:2C02
IPv6 Gateway
IPv6 DNS Server
802.1X
Use 802.1X Security
Authentication
MD5
Username
Top

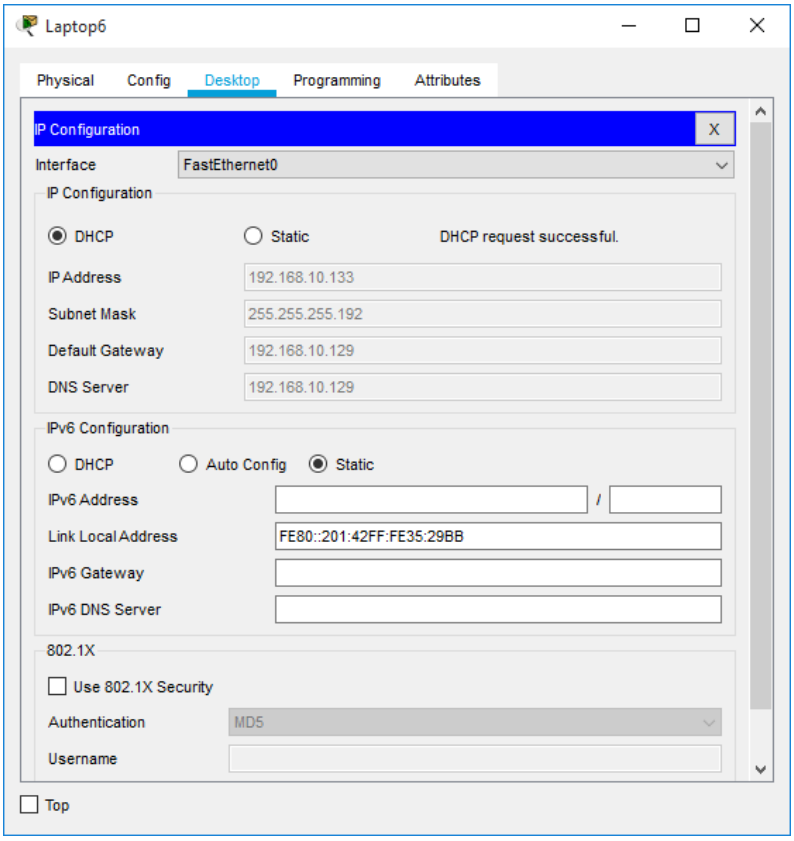
в)


Laptop4
Physical
Config
Desktop
Programming
Attributes
IP Configuration
X
Interface
FastEthernet0
IP Configuration
DHCP
Static
DHCP request successful.
IP Address
192.168.10.69
Subnet Mask
255.255.255.192
Default Gateway
192.168.10.65
DNS Server
192.168.10.65
IPv6 Configuration
DHCP
Auto Config
Static
IPv6 Address
Link Local Address
FE80::2E0:F7FF:FE24:8A75
IPv6 Gateway
IPv6 DNS Server
802.1X
Use 802.1X Security
Authentication
MD5
Username
Top

г)



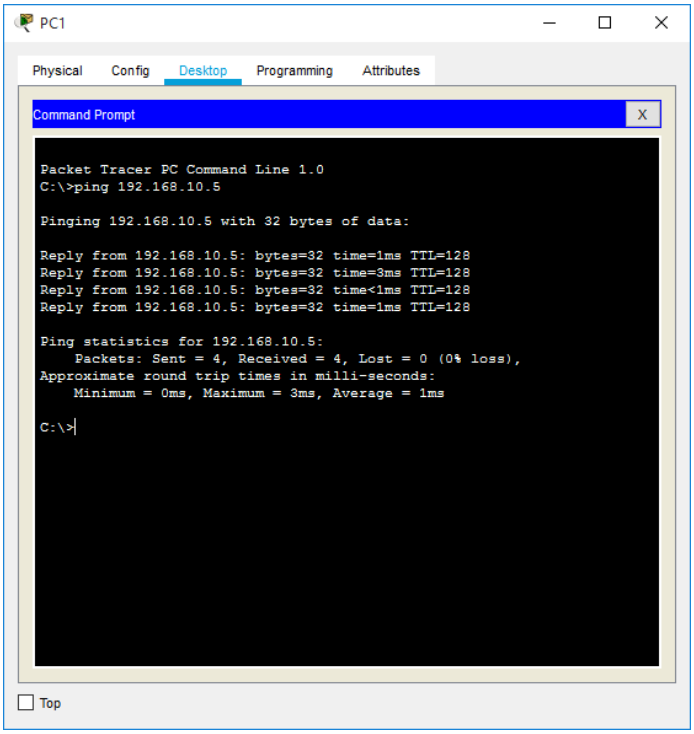
д)



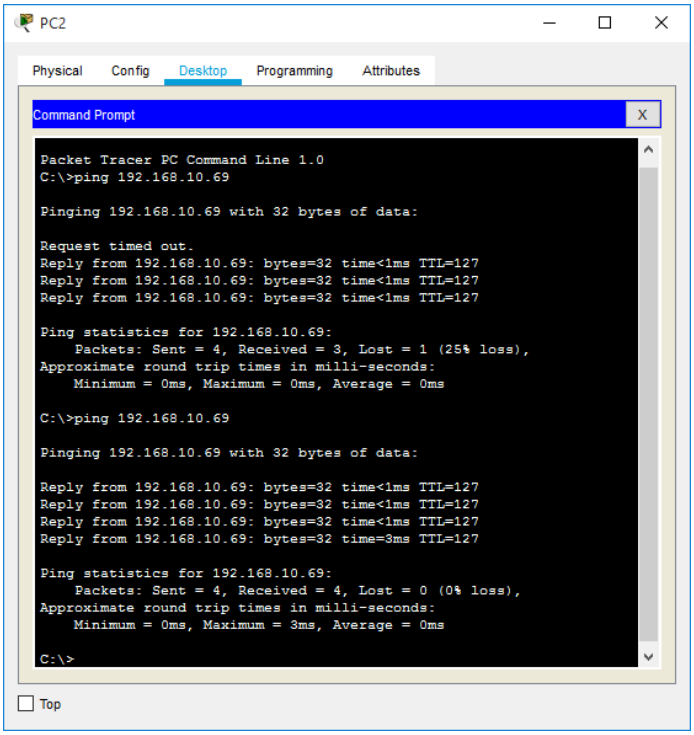
е)

Фиг. 4.12. *DHCP процедури при PC1 а), Laptop2 б), PC3 в), Laptop4 г) PC5 д) и Laptop6 е)*

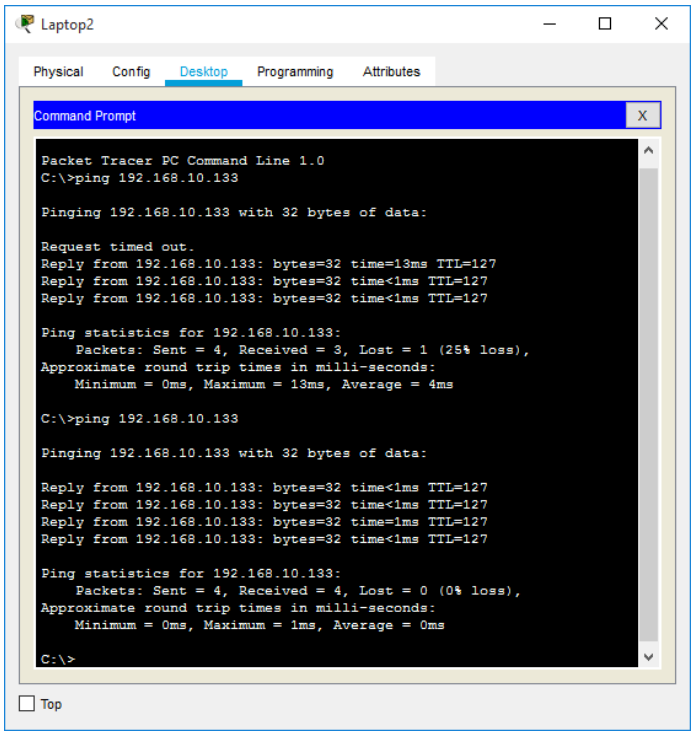
На последен етап се преминава към тестване на работоспособността на разглежданата мрежова конфигурация по отношение на пакетното предаване на данни между мрежовите устройства в рамките на една помрежа - фиг. 4.13, и между отделните сегменти, както е показано на фиг. 4.14. Действията се състоят в извършване на Ping процедура в Command Prompt интерфейс в режим Reaaltime.



Фиг. 4.13. *Ping процедура от PC1 към Laptop2 в Realtime режим*

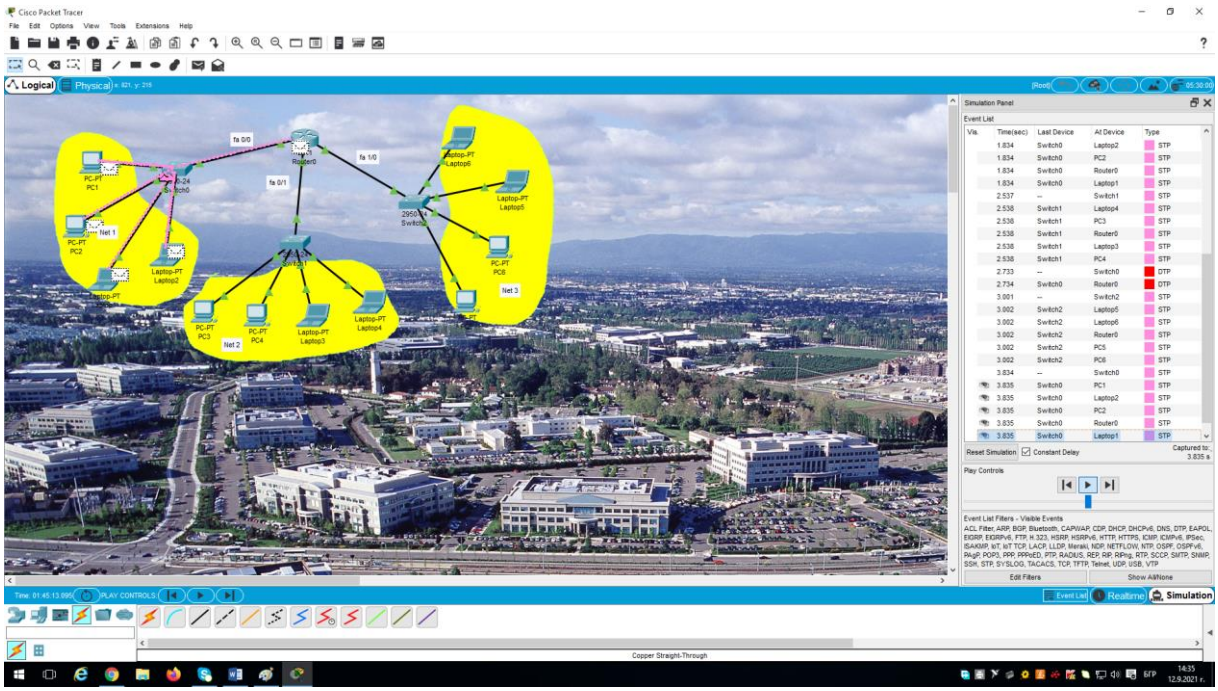


а)



б)

Фиг. 4.14. *Ping процедури в Realtime режим а) от PC2 от Net №1 към Laptop4 от Net №2 и б) от Laptop2 от Net №1 към Laptop6 от Net №3*



Фиг. 4.15. *Изпращане на пакети в режим Simulation*

Фигура 4.15 представя хода на пакетен информационен трансфер между персонални мрежови станции в Simulation mode. Установени са процеси на успешно изпращане и получаване на всички единици пакети от данни, т.е. предаването на PDU информационни единици се изпълнява успешно между мрежовите устройства от различните подмрежи.

4.4. Контролни въпроси

1. В кои работни среди и режими на работа се проектират компютърните конфигурации и провежда пакетен мониторинг между мрежовите устройства?
2. В какви ситуации се прилага Straight-trought тип на свързване между мрежовите устройства?
3. Между кои двойки мрежови устройства се използва Crossover тип на връзката?
4. В какво се състои процедурата по IP конфигурация на мрежовите устройства?
5. Кои режими на назначаване на IP адреси се прилагат в мрежовите устройства?
6. Кое определя обхвата на даден мрежови сегмент по отношение на максималното количество устройства, които могат да бъдат включени в неговата структура?

Лабораторно упражнение № 5

МОДЕЛИРАНЕ И КОНФИГУРИРАНЕ НА БЕЗЖИЧНИ КОМПЮТЪРНИ КОНФИГУРАЦИИ ЧРЕЗ СИМУЛАТОРА CISCO PACKET TRACER

5.1. Цел на упражнението

Да се запознаят студентите с основните режими за обезпечаване на сигурността и оторизацията на потребителския достъп в wireless мрежови конфигурации и включването на wireless сегменти чрез безжична точка за достъп към жични структури.

5.2. Теоретична постановка

През последните години все повече нараства внедряването на безжичните мрежи и технологии при проектиране на информационно-комуникационни трасета. Техните предимства се свързват с „удобство“, „мобилност“, „продуктивност“, „разширяемост“, „цена“ и др. Недостатъците им се изразяват в по отношение появата на нарушения в „сигурността“, „надеждността“, „обхвата“, „скоростта“ и „консумираната енергия и отделяната топлина“. Съществува многообразие от безжични стандарти при изграждане на LAN архитектури, различаващи се според поддържана „максимална и реална скорост“, „брой на каналите“, „максимално покритие“, „носеща честота“ и т.н.

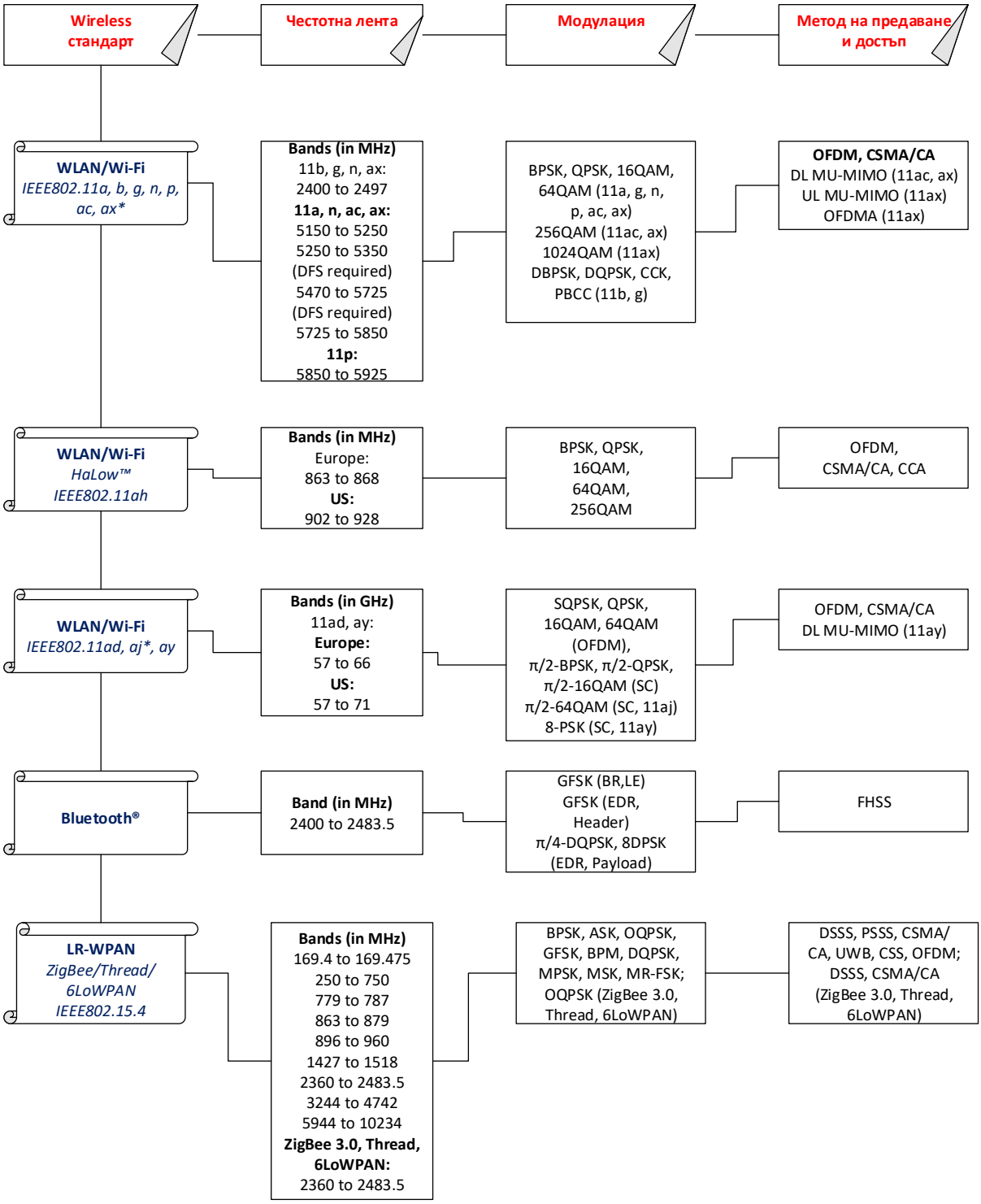
Безжичните мрежи се разделят на два вида:

- Мрежи тип *ad-hoc (peer-to-peer)* - Терминът ad-hoc означава “специален”, а peer-to-peer “равен с равен”;
- Мрежи с използване на точки за достъп (*AP – Access Point*) до жични LAN мрежови топологии.

На фиг. 5.1 са представени Wireless стандарти, както и конкретни специфицирани данни по отношение на „честотната лента“, поддържаните „типове модулации“ и „методи на предаване на данни и достъп до комуникационните канали“. Някои от основните протоколи относно „контрол на достъпа“ и „маршрутизацията“ са обобщени на фиг. 5.2 [9].

Фигура 5.3 дефинира режимите, свързани с управление на сигурността при защита на данните в Wireless мрежите. В случая са показани различни опции за избор на режим в процесите на настройка на сигурността при мрежови „персонални потребителски компютърни станции“, „безжични рутери“ и устройства от тип „точка за достъп“ при симулационно моделиране на конфигурации в пакета Cisco Packet Tracer. В практиката се използват четири основни начина за кодиране на данните:

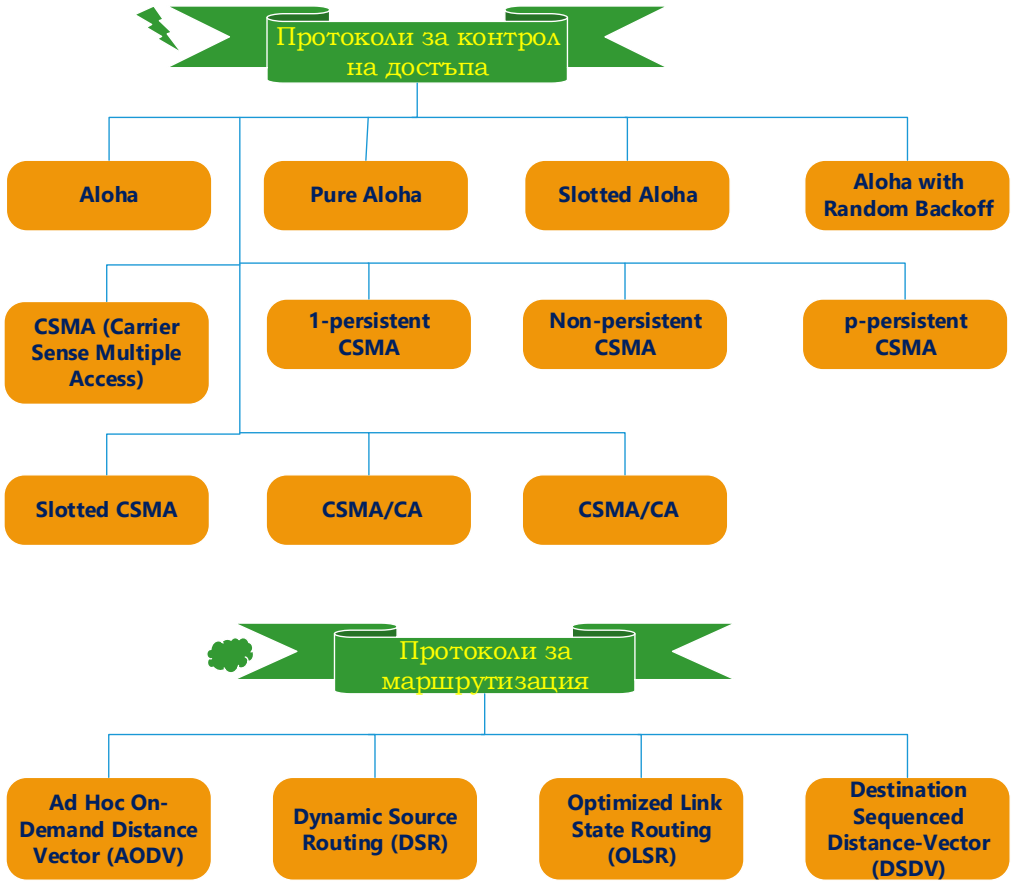
- *WEP* (Wired Equivalency Privacy) – при WEP се използва RC4 алгоритъм за криптиране с дължина на ключовете 64, 128 или 256 бита, притежавани от всички мрежови устройства в конкретната конфигурация. Използват се HEX, ASCII и други типове криптиращи ключове. По-високо ниво за защита се постига при WPA-PSK криптиране;



Фиг. 5.1. *Базови Wireless стандарти в комуникационните системи*

- *WPA* (Wi-Fi Protected Access) – поддържа TKIP (Temporal Key Integrity Protocol) и AES (Advanced Encryption Standard) подходи за криптиране при до 256 битов ключ, който се променят периодично. Съществуват две версии на мрежова автентикация "WPA Enterprise" и "WPA Personal" за корпоративни и частни бизнес мрежи. При първата версия се прилага EAP (Extensive Authentication Protocol) автентикация, докато при WPA personal съответен „споделен ключ" (PSK – Pre-Shared Key) във връзка със създаване на фраза за

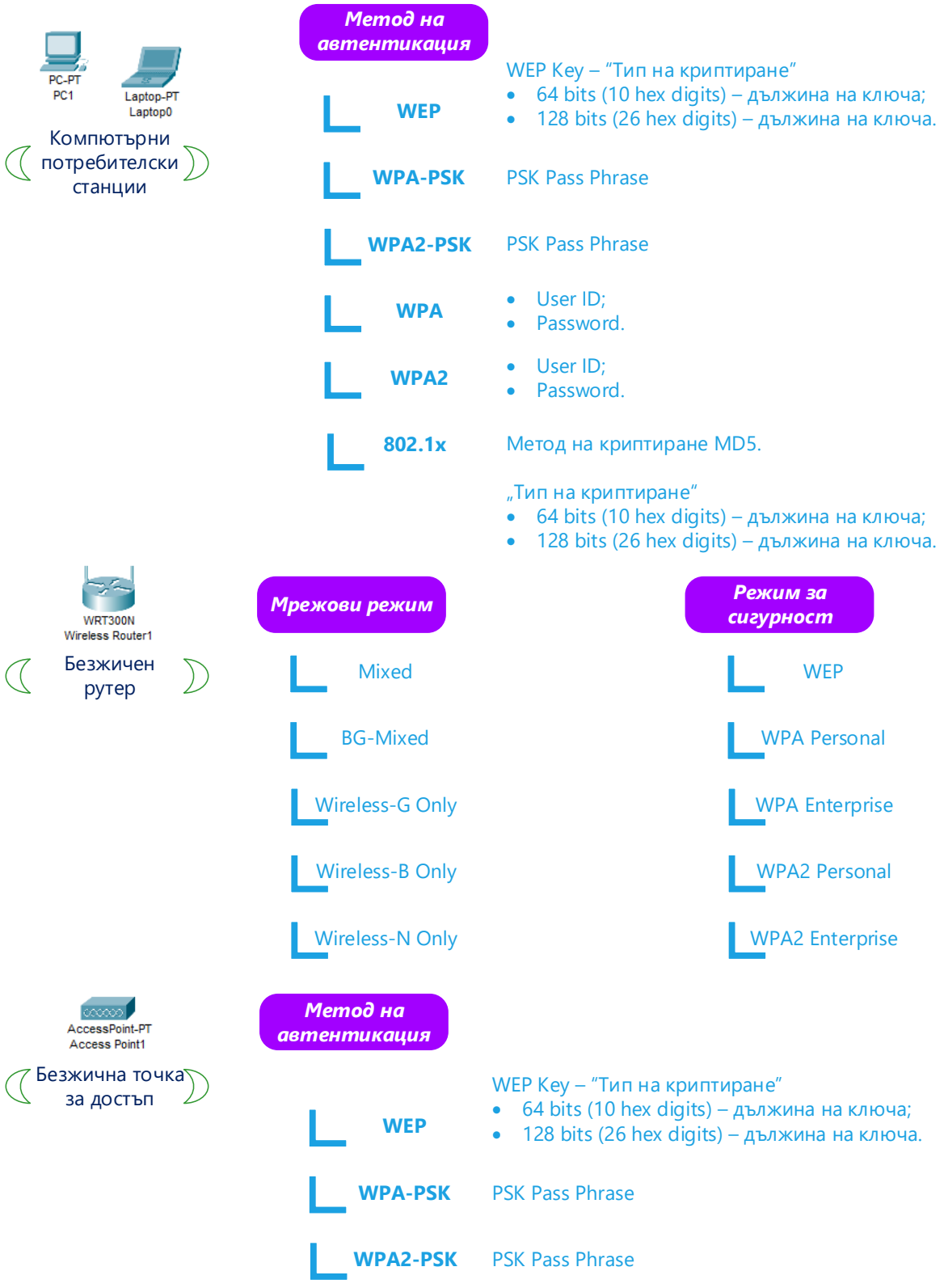
защита от 8 до 63 символа. WPA криптиране в съчетание с PSK е известно като WPA-PSK;



Фиг. 5.2. *Протоколи за контрол на достъпа и маршрутизация в Wireless мрежови инфраструктури*

- *WPA2* криптиране – представлява най-новата версия на стандарта 802.11i. Предимство спрямо WAP режима за защита е в регламентирането на AES като основен алгоритъм. Този подход се използва и в двата варианта „WPA2 Enterprise“ и “WPA2 Personal” при процедурно дефиниране на сигурността;

- Назначаване на *SSID* персонализиращи идентификатори – свързва се с процесите на контрол на достъпа на безжичните устройства или точките за достъп (Aps) при LAN и WLAN чрез присвояване на уникално „идентификационно име“ SSID (Service Set ID). Всички налични устройства в обхвата на дадена мрежа използват един и същ, докато припокриващите по обхват различни SSID. Освен назначения идентификатор всяка AP съхранява и списък с MAC адресите на мрежовите устройства, с които е упълномощена да комуникира [10].

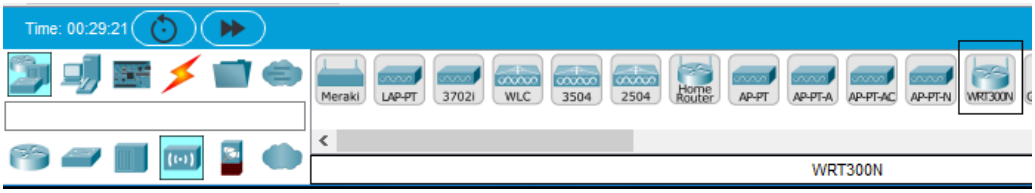


Фиг. 5.3. *Режими за управление на сигурността относно персонални потребителски станции, безжични рутери и точки за достъп, поддържани в симулационна среда Cisco Packet Tracer*

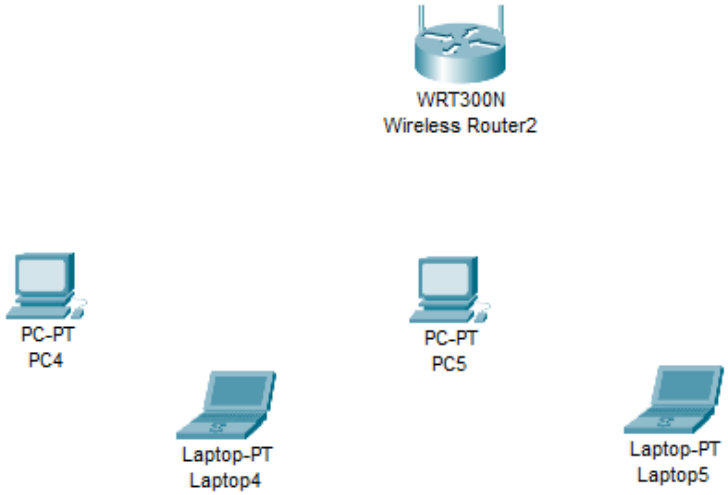
5.3. Последователност на работа и задачи за изпълнение

5.3.1. Конфигуриране на безжична комуникация между безжичен рутер и персонални потребителски станции

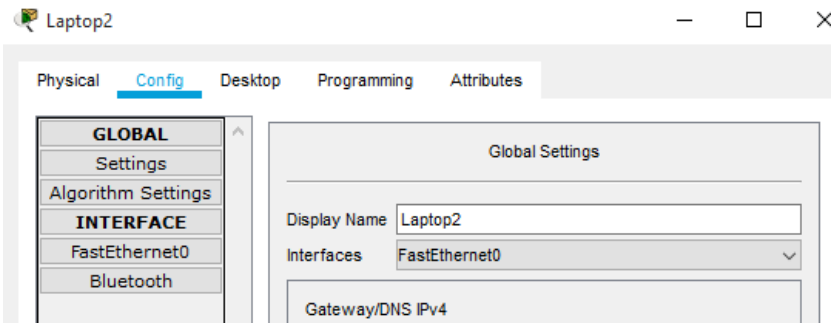
Използването на безжична връзка за изграждане на мрежови сегменти и обмяна на потребителски данни може да се илюстрира чрез използването на безжичен рутер WRT300N, който е показан на фиг. 5.4. Фигура 5.5 представя примерна конфигурация на стационарни и преносими потребителски станции, които следва да бъдат свързани и конфигурирани за комуникация с посочения тип безжичен рутер. Както може да се види от фиг. 3.10, структурните мрежови устройства PC1, Laptop2, PC3 и Laptop4 не разполагат с wireless интерфейс, с помощта на който да се осъществи безжична връзка с рутер WRT300N.



Фиг. 5.4. *Достъп до безжичен рутер WRT300N в Cisco Packet Tracer*



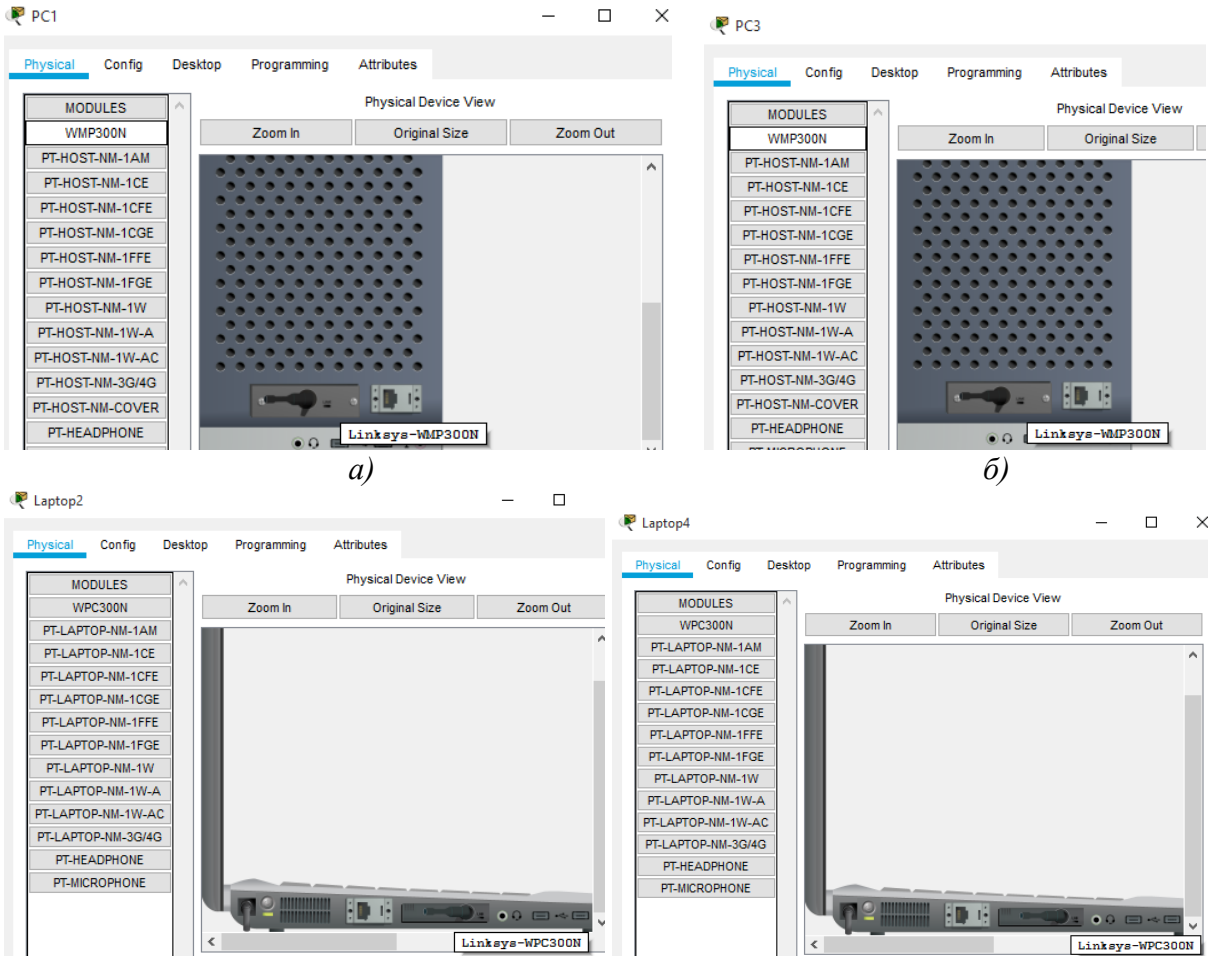
Фиг. 5.5. *Конфигурация с включване на персонални потребителски станции за връзка с безжичен рутер WRT300N*



Фиг. 5.6. *Налични интерфейси преди добавяне на модула за wireless интерфейс при персонална преносима станция Laptop2*

Ето защо е необходимо добавянето на модул за тази цел с включването, на който се реализира от конфигурационен раздел Physical, извеждащ налични категории интерфейсни модули за различни типове свързвания между мрежови устройства. Съобразно WRT300N се използват многофункционални модули WMP300N за стационарни и WPC300N за преносими компютърни станции. Добавянето на модулите се извършва при първоначално изключване на захранването на конкретна потребителска станция и отстраняване на модул PT-HOST-NM-1CFE за PC1 и PC3, както и PT-LAPTOP-NM-1CFE при Laptop2 и Laptop4.

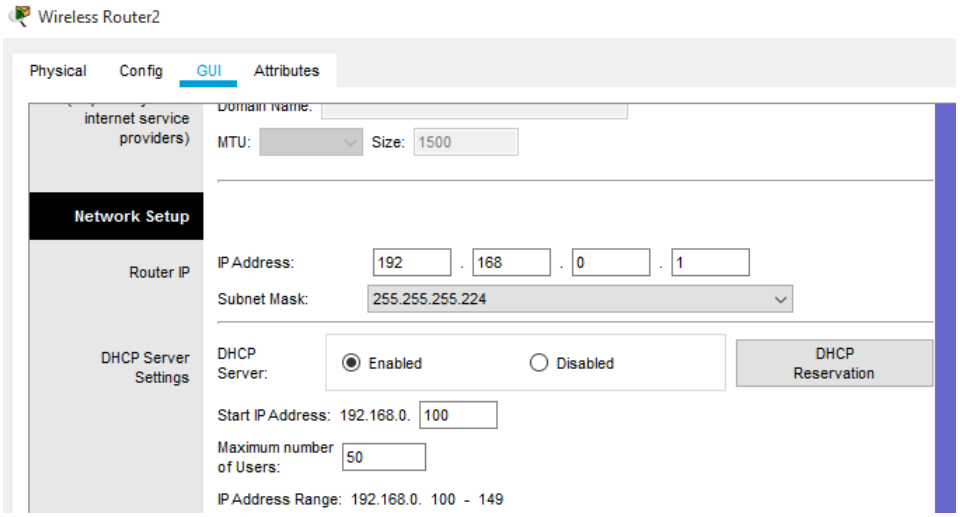
Указаните интерфейсни модули осигуряват единичен Fast-Ethernet порт чрез меден проводник. Необходимите безжични модули WMP300N и WPC300N се поставят в съответните гнезда, след което захранването на станциите отново се възстановява. Посочените дейности са представени на фиг. 5.7.



Фиг. 5.7. *Добавяне на модули за безжична комуникация WMP300N за стационарни PC1 а) и PC3 б) и WPC300N за преносими станции Laptop2 в) и Laptop4 г)*

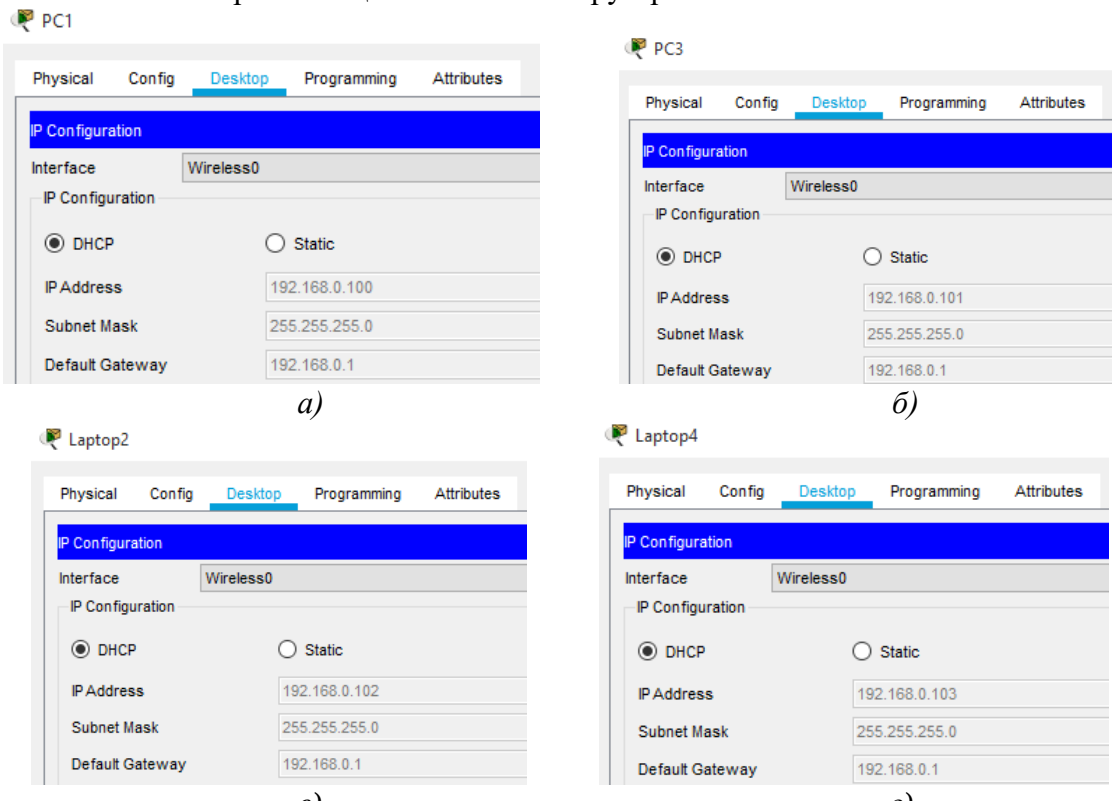
Непосредствено с това се извършва проверка относно коректността на назначаване на динамичните конфигурационни настройки – тип комуникационен интерфейс, унифициращ мрежови IP адрес, подмрежова маска и Gateway (Шлюз), при станции PC1, Laptop2, PC3 и Laptop4. По отношение на процедурата при стартиране на графичния потребителски интерфейс (GUI) на безжичения рутер WRT300N на фиг. 5.8 предварително могат да прегледани основните мрежови настройки във връзка с рутера (IP адрес, Subnet mask) и DHCP сървъра:

- разрешаване/забраняване на опцията за динамично назначаване на IP адреси;
- начален мрежови IP адрес;
- максимален брой потребители;
- обхват от IP адреси за потенциално назначаване.

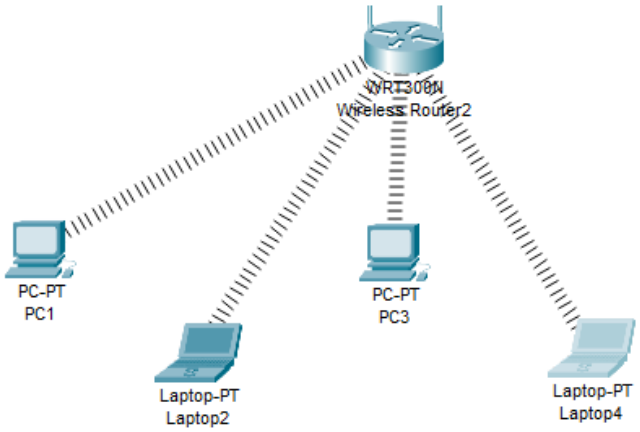


Фиг. 5.8. *Начални конфигурационни настройки на безжичен рутер WRT300N, достъпни през GUI модул*

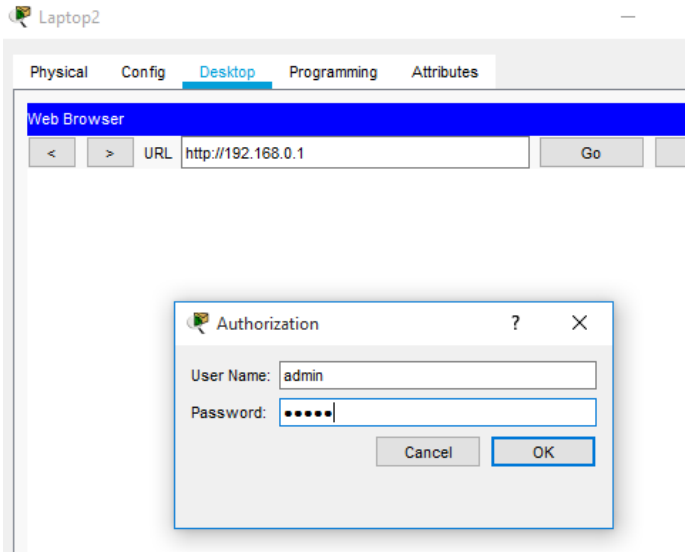
Фигура 5.9 онагледява коректното и успешно назначаване на DHCP настройки (избор на раздел Desktop -> инструмент IP Configuration), а фиг. 5.10 свързване между мрежовите компютърни станции и безжичния рутер.



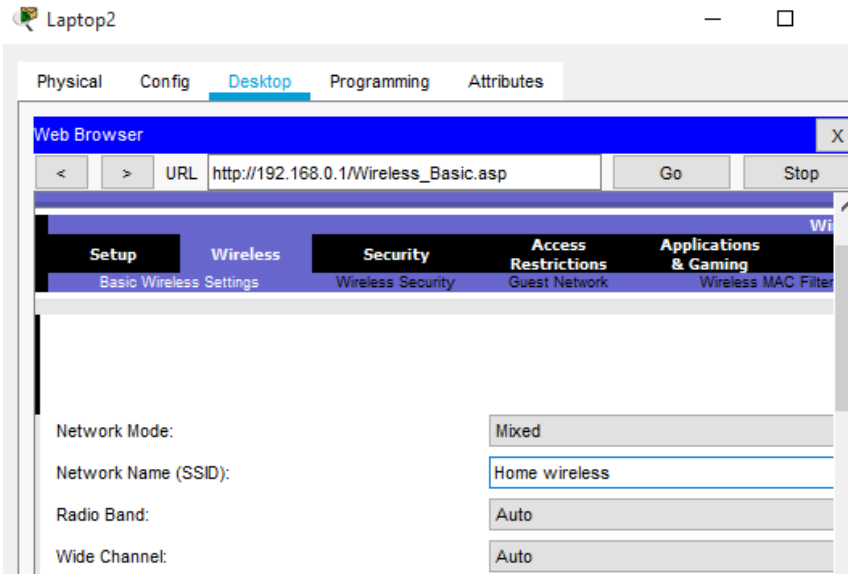
Фиг. 5.9. *Коректност на DHCP конфигурацията при компютърни станции а) PC1, б) PC3, в) Laptop2 и г) Laptop4*



Фиг. 5.10. *Изглед на мрежовата конфигурация преди оказване на режим за сигурността относно мрежовата автентикация*

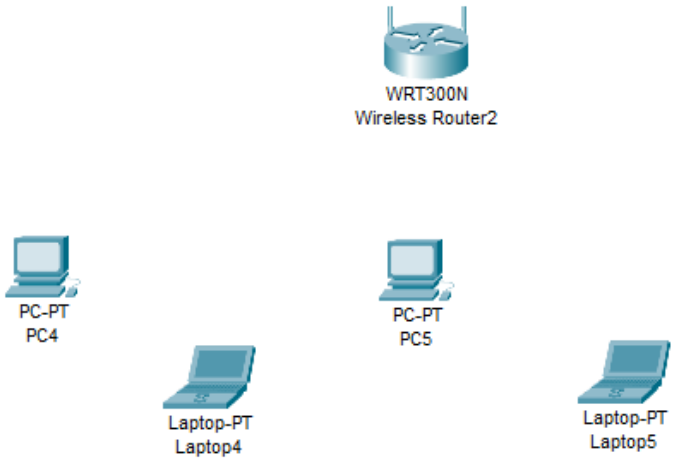


а)



б)

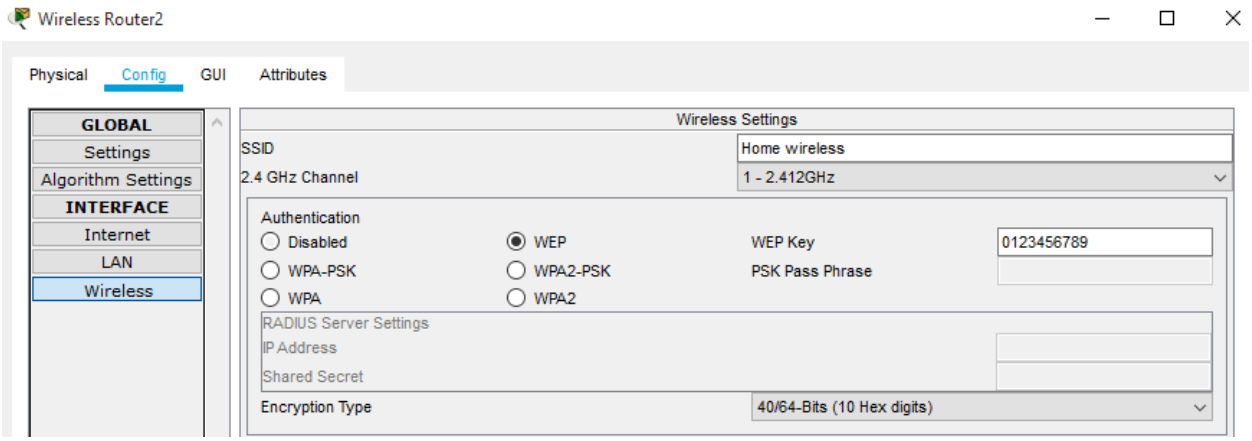
Фиг. 5.11. *Достъп до GUI на безжичен рутер WRT300N през WEB Browser на компютърна станция Laptop2 – достъп до GUI а) и назначаване на мрежови SSID б)*



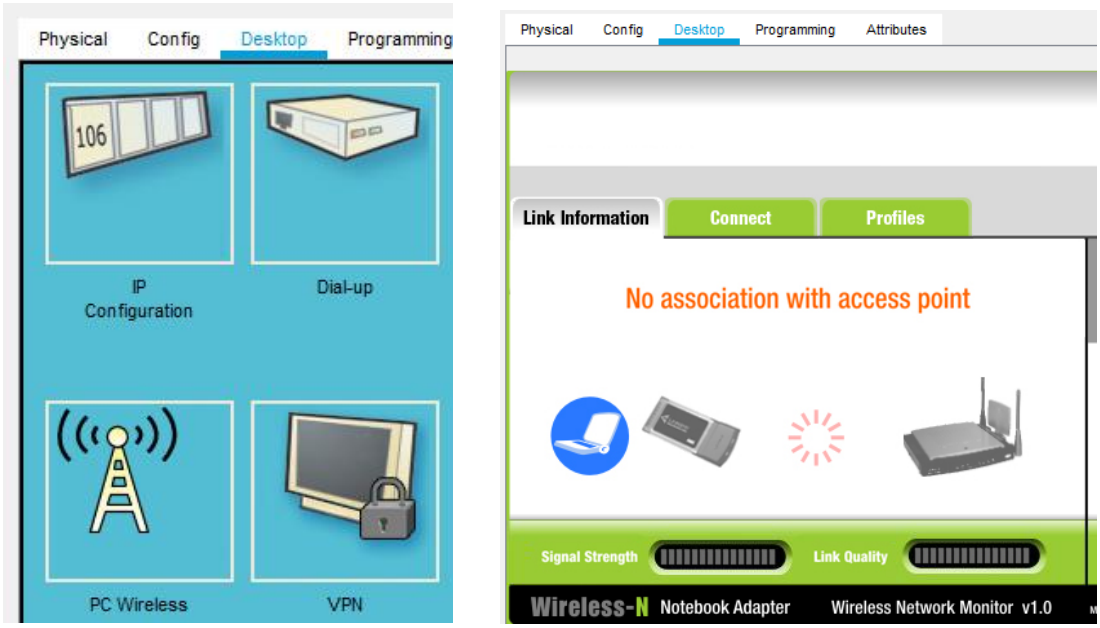
Фиг. 5.12. *Преустановяване на връзката между мрежовите устройства след назначаване на нов потребителски мрежови SSID*

Следващата фаза от провеждания конфигурационен режим се състои в обезпечаване на сигурността и потребителския достъп относно информационния обмен. Тук на първо място се задава идентификатор на безжичната мрежа SSID “Home wireless” в раздел „Wireless“ на GUI инструмента на WRT300N. В случая потребителският интерфейс може да бъде достъпен през стандартно WEB Browser приложение, от която и да било стационарна или преносима компютърна станция, чрез въвеждане на неговия адрес 192.168.0.1 и администраторско име и парола за достъп “admin” на фиг. 5.11. При този процес следва преустановяване на връзката между мрежовите устройства (фиг. 5.12).

След това се извършва избор на „*Режим на автентикация*“ *WEP* (*Wired Equivalency Privacy*) и задаване на *WEP Key “0123456789”* относно защита на безжичната връзка. Техническият способ за сигурност WEP използва RC4 алгоритъм за криптиране с дължина на ключовете 64, 128 или 256 бита, притежавани от всички мрежови устройства в конкретната конфигурация. Използват се HEX, ASCII и други типове криптиращи ключове. Тип на криптиране 64 bits се отнася до употреба на 10 hex digits, докато 128 bits съответно 24 hex digits. Указаната процедура с прилагане на 64 bits дължина на ключа се провежда в конфигурационен раздел Config относно интерфейса “Wireless” на фиг. 5.13.



Фиг. 5.13. *Задаване на режим за сигурност WEP с WEP Key “0123456789”*

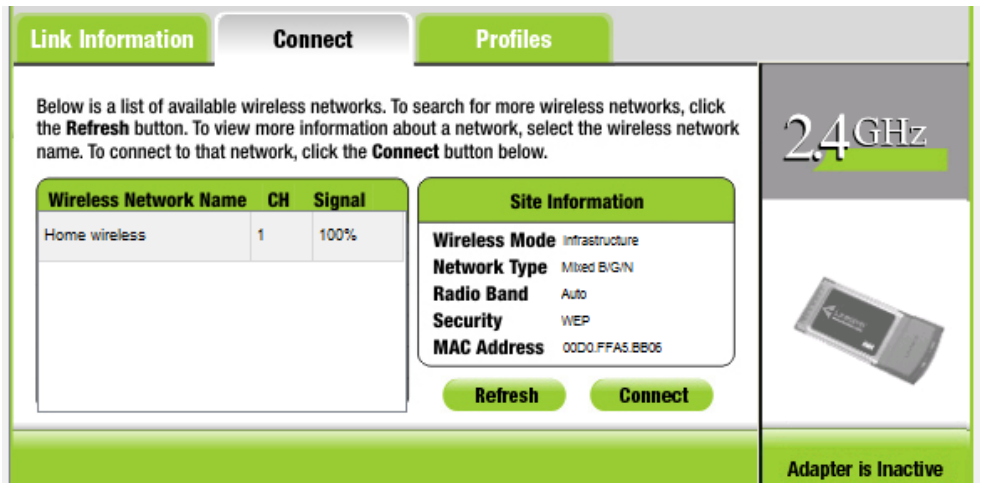


а) *б)*

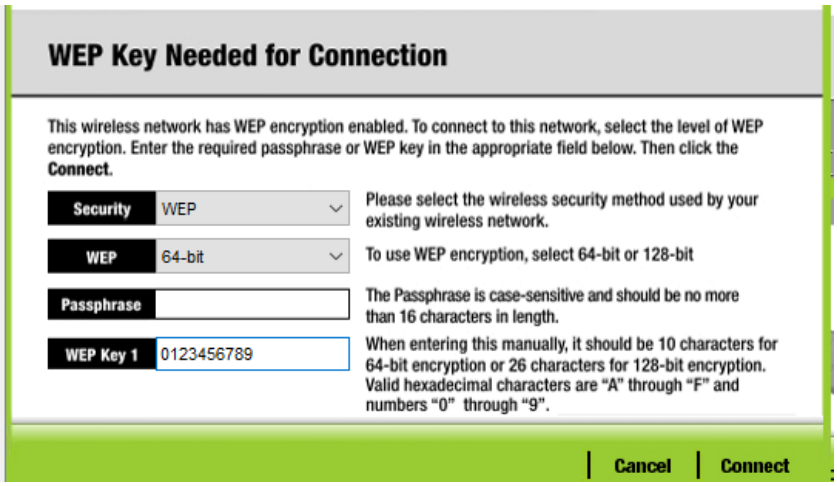
Фиг. 5.14. *Приложение PC Wireless в раздел Desktop а) и конфигурационен прозорец за настройка на безжичния адаптер б) при потребителските компютърни станции*

За да бъде осъществена връзка между стационарните и преносимите компютърни станции с безжични рутер WRT300N спрямо избрания режим на автентикация, е необходимо да бъдат въведени идентични WEP настройки при всяко мрежово устройство. Съгласно това се използва приложение PC Wireless и избор на конфигурационен диалогов прозорец Connect за активация на *безжичните 2.4 GHz мрежови адаптери* на персоналните компютърни станции на фиг. 5.14. Процедурите се свеждат до:

- избиране на конфигурационен раздел Connect и обновяване на статуса чрез опция Refresh относно извеждане на наличните безжични мрежи и параметрична спецификация на връзката (мрежови идентификатор, комуникационен канал за връзка, ниво на поддържания сигнал, безжичен режим, режим на сигурност, назначен регулярен диапазон на подържаната честотна лента, MAC адрес), представено на фиг. 5.15;



Фиг. 5.15. *Извеждане на наличните безжични мрежи и специфична параметрична информация*



Фиг. 5.16. *Конфигурация на параметрите на WEP режим при съответна дължина на ключа*

- избиране на опция Connect и настройка на параметрите на WEP режим на автентикация, състояща се във въвеждане на унифициран WEP Key 1 "0123456789" при 64 bits тип на криптиране и повторно избиране на опцията Connect, дадено на фиг. 5.16;

- удостоверяване на коректността на активация на безжичния мрежови адаптер и статуса на безжичната връзка между конкретното мрежово устройство и рутер WRT300N, показано на фиг. 5.17 и фиг. 5.18;

а)


PC3
Physical
Config
Desktop
Programming
Attributes
Link Information
Connect
Profiles
More Information
Infrastructure Mode
2.4GHz
You have successfully connected to the access point
Signal Strength
Link Quality
Adapter is Active
Wireless-N Notebook Adapter
Wireless Network Monitor v1.0
Model No. WPC300N

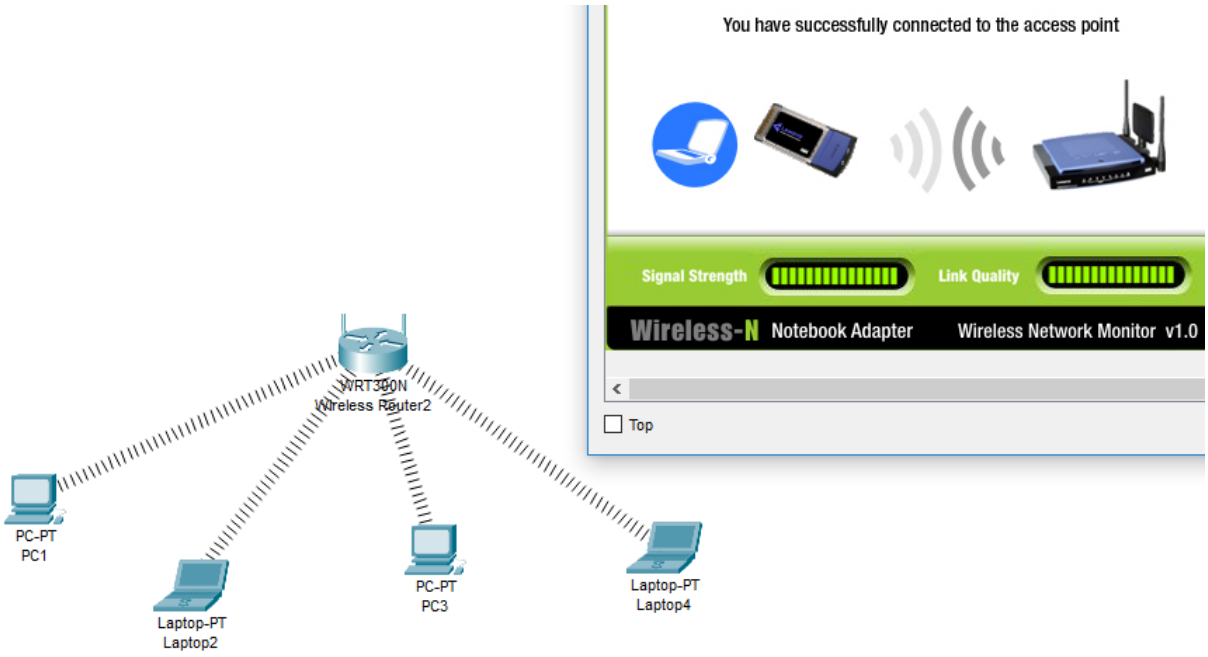
б)


Laptop2
Physical
Config
Desktop
Programming
Attributes
Link Information
Connect
Profiles
More Information
Infrastructure Mode
2.4GHz
You have successfully connected to the access point
Signal Strength
Link Quality
Adapter is Active
Wireless-N Notebook Adapter
Wireless Network Monitor v1.0
Model No. WPC300N

в)

г)

Фиг. 5.17. *Активация на безжичните мрежови адаптери при персонални компютърни станции а) PC1, б) PC3, в) Laptop2 и г) Laptop4*



Фиг. 5.18. *Успешно свързване между мрежовите устройства в безжичната инфраструктура*

- тестване на пакетното предаване на PDU (Personal Data Unit) данни между стационарните и преносимите потребителски станции в режим Realtime, използвайки ping процедури в основната работна среда и приложение Command prompt , както е илюстрирано на фиг. 5.19;

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	PC1	Laptop2	ICMP		0.000	N	0	(edit)	(delete)
	Successful	PC1	PC3	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC1	Laptop4	ICMP		0.000	N	2	(edit)	(delete)

а)

PC1

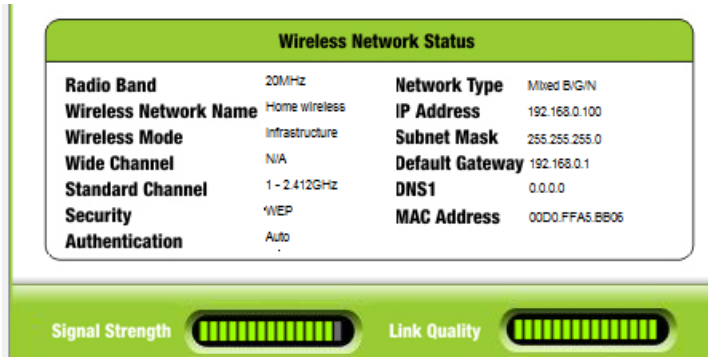
Physical Config Desktop Programming Attributes

Command Prompt

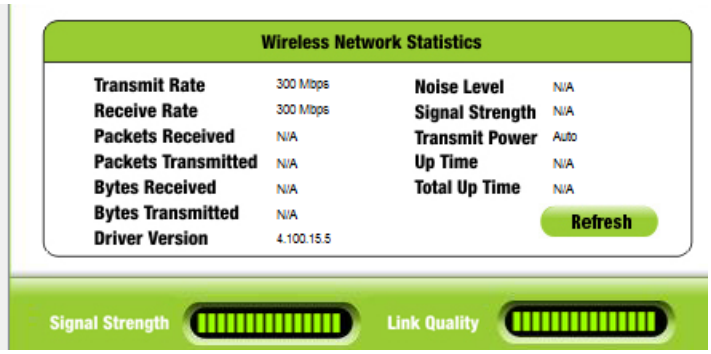
```
Packet Tracer PC Command Line 1.0
C:\>

C:\>ping 192.168.0.102

Pinging 192.168.0.102 with 32 bytes of data:

Reply from 192.168.0.102: bytes=32 time=20ms TTL=128
Reply from 192.168.0.102: bytes=32 time=35ms TTL=128
Reply from 192.168.0.102: bytes=32 time=28ms TTL=128
Reply from 192.168.0.102: bytes=32 time=26ms TTL=128

Ping statistics for 192.168.0.102:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 20ms, Maximum = 35ms, Average = 27ms

C:\>ping 192.168.0.101

Pinging 192.168.0.101 with 32 bytes of data:

Reply from 192.168.0.101: bytes=32 time=24ms TTL=128
Reply from 192.168.0.101: bytes=32 time=30ms TTL=128
Reply from 192.168.0.101: bytes=32 time=9ms TTL=128
Reply from 192.168.0.101: bytes=32 time=18ms TTL=128

Ping statistics for 192.168.0.101:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 9ms, Maximum = 30ms, Average = 20ms

C:\>ping 192.168.0.103

Pinging 192.168.0.103 with 32 bytes of data:

Reply from 192.168.0.103: bytes=32 time=26ms TTL=128
Reply from 192.168.0.103: bytes=32 time=19ms TTL=128
Reply from 192.168.0.103: bytes=32 time=24ms TTL=128
Reply from 192.168.0.103: bytes=32 time=20ms TTL=128

Ping statistics for 192.168.0.103:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 19ms, Maximum = 26ms, Average = 22ms
```

б)

Фиг. 5.19. *Ping процедури по направление от PC1 към станции Laptop2, PC3 Laptop4 в а) основната работна среда и б) Command Prompt приложение*



а)



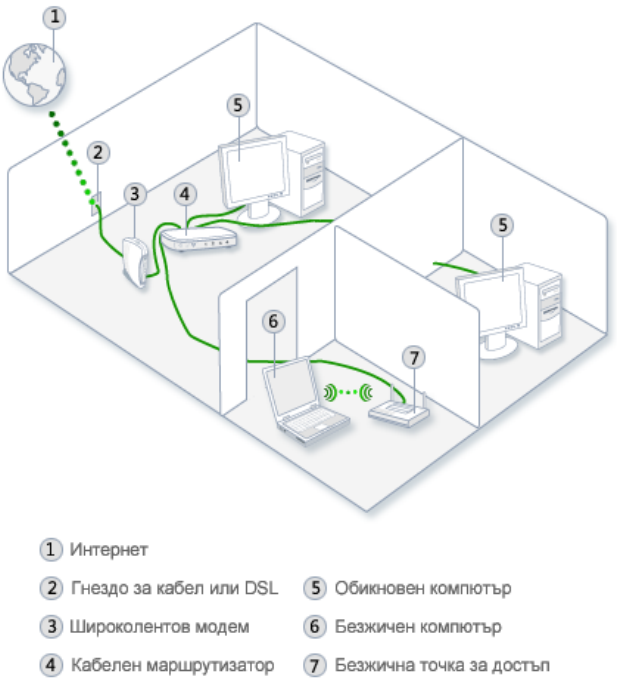
б)

Фиг. 5.20. *Текущ статус на безжичната връзка а) и статистически индикатори за трафика б) при персонална компютърна станция PC3*

- проследяване и анализиране на текущият параметричен статус на мрежовата връзка и статистическите показатели (скорост и количество на изпратените/получените пакети, обем на предаваните/получаваните данни, ниво на шума, сила на сигнала и т.н.) за информационния трафик между мрежовите устройства, представено на фиг. 5.20.

5.3.2. Моделиране на комбинирана жична и безжична инфраструктура с включване на устройства посредством Access Point в Cisco Packet Tracer

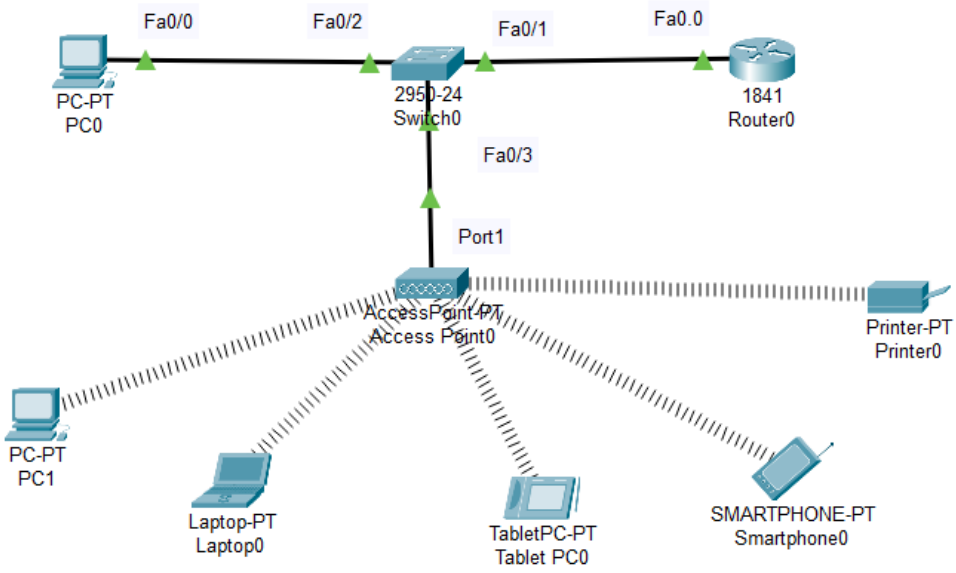
Стандартните безжични локални мрежи изглеждат и работят точно като кабелните мрежи, с изключение на физическата среда. Във всеки компютър се инсталира безжична мрежова адаптерна карта с приемо-предавателно устройство, което е известно с названието „**трансийвър“** (transceiver), а потребителите комуникират по същия начин, както ако бяха свързани чрез жична връзка. Трансийвърите, наричани понякога още „**точки на достъп“** или **Aaccess Points (AP),** излъчват и получават сигнали от заобикалящите ги компютърни станции, като ги препредават по безжична или кабелна среда. Безжичните локални мрежи използват за връзка с кабелните мрежи малки трансийвъри, които обикновено са монитрани в стените. AP звената установяват радио-контакт с портативните мрежови устройства. Такава инфраструктура не е изцяло безжична LAN, защото се използват монтирани в стената трансийвъри за връзка със стандартната кабелна LAN, представено на фиг. 5.21.



Фиг. 5.21. *Концепция за функционална мрежова Access Point*

Симулационната среда Cisco Packet Tracer позволява моделиране и мониторинг на трафика в мрежови инфраструктури с включване на Access Point. Сценарий за подобна архитектура е даден на фиг. 5.22. Представената WLAN топология се състои от два мрежови сегмента:

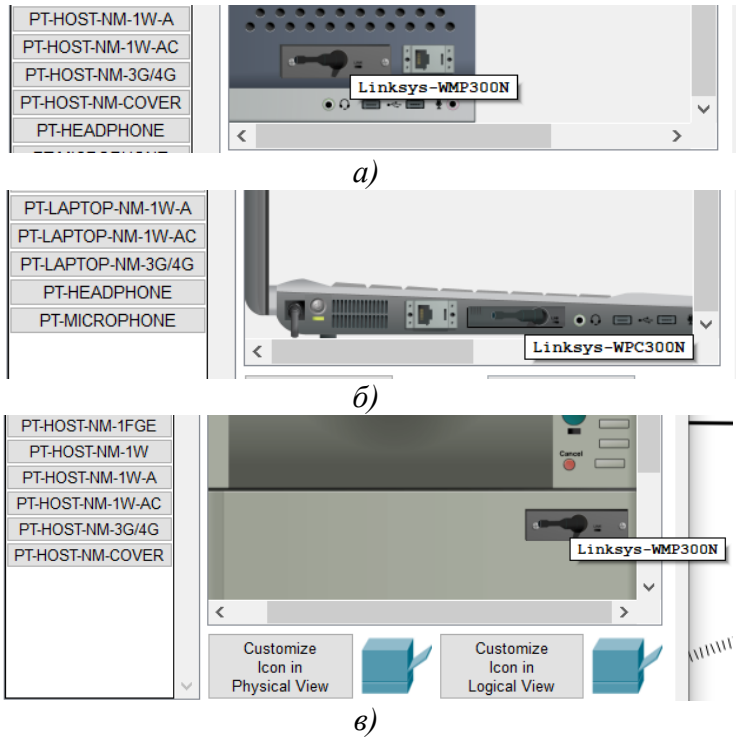
- „жичен“, съдържаща персонална станция PC0, комутатор и маршрутизарор типове Switch 2950-24 Router 1841 и мрежова „точка за достъп“ Access Point-PT. Връзката между споменатите устройства се реализира чрез използването на Fast Ethernet тип интерфейс, както Fast Ethernet от страна на комутатора към Port1 на „точката за достъп“;



Фиг. 5.22. *Конфигурация с включване на мрежови устройства с Access Point в Cisco Packet Tracer*

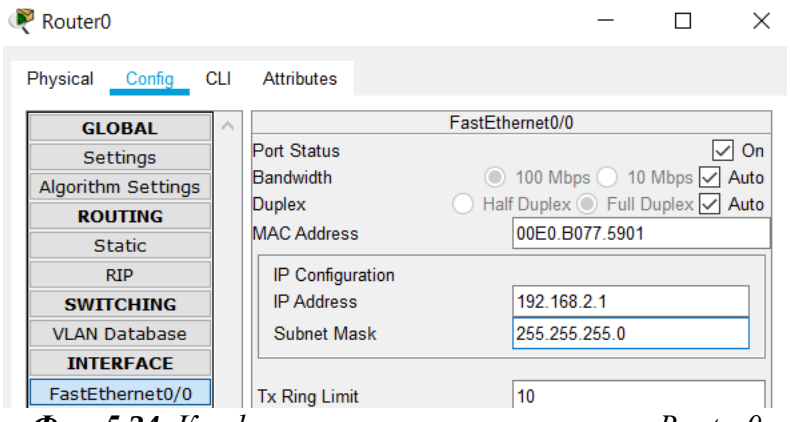
- „безжичен“, който включва персонални компютърни станции PC1 и Laptop0, интелигентни безжични преносими устройства TabletPC-PT и SMARTPHONE-PT, също така и мрежови принтер Printer-PT. Тук именно се осъществява връзката между указаните устройства и специализираното мрежово звено Access Point.

За да бъде реализирана безжична връзка между Access Point и мрежовите устройства PC0, Laptop0 Printer0, към всяко от устройствата трябва да бъде добавен специализиран модул WMP300N/ WPC300N, както е посочено на фиг. 5.23. След изграждане на физическите връзки следва да се премине към конфигурация на мрежовите устройства.

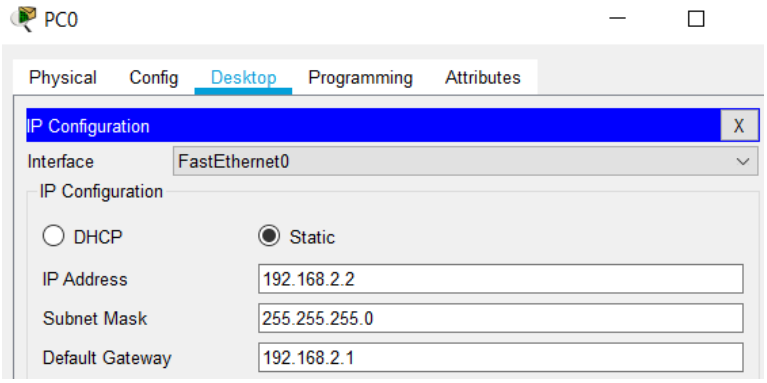


Фиг. 5.23. *Добавяне на модули за безжична връзка а) Linksys-WMP300N при PC0, б) Linksys-WPC300N при Laptop0 и в) Linksys-WMP300N при Printer0*

На фиг. 5.24 е онагледена процедурата по задаване на IP адрес и подмрежова маска на маршрутизатора Router0 по отношение на FastEthernet0/0. Във връзка с персонална потребителска станция PC0 се извършва назначаване на IP адрес, подмрежова маска и Gateway по подразбиране, както е показано на фиг. 5.25.

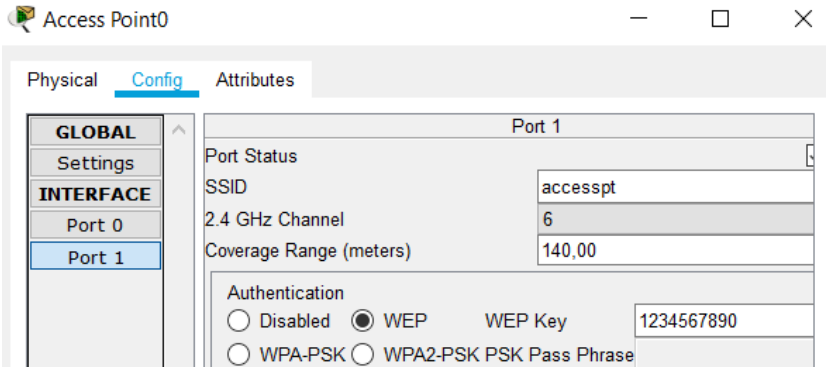


Фиг. 5.24. *Конфигурация на маршрутизатор Router0*

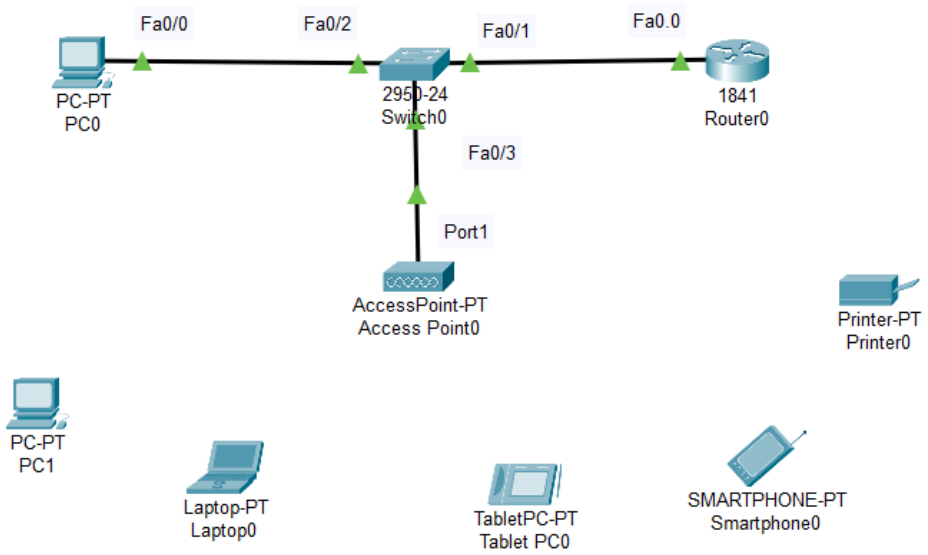


Фиг. 5.25. *Настройки на персонална станция PC0*

Следващата стъпка от дейностите по конфигурация на „**точката за достъп**“ Access Point се отнася до задаване на индентификатор на безжичната мрежа SSID (Service Set Identifier) “accesspt” и режим за сигурност/автентикация (удостоверяване) WEP (Wired Equivalent Privacy) с криптиращ ключ “1234567890”. Процедурата се извършва по отношение на интерфейсен порт Port1, показано на фиг. 5.26.

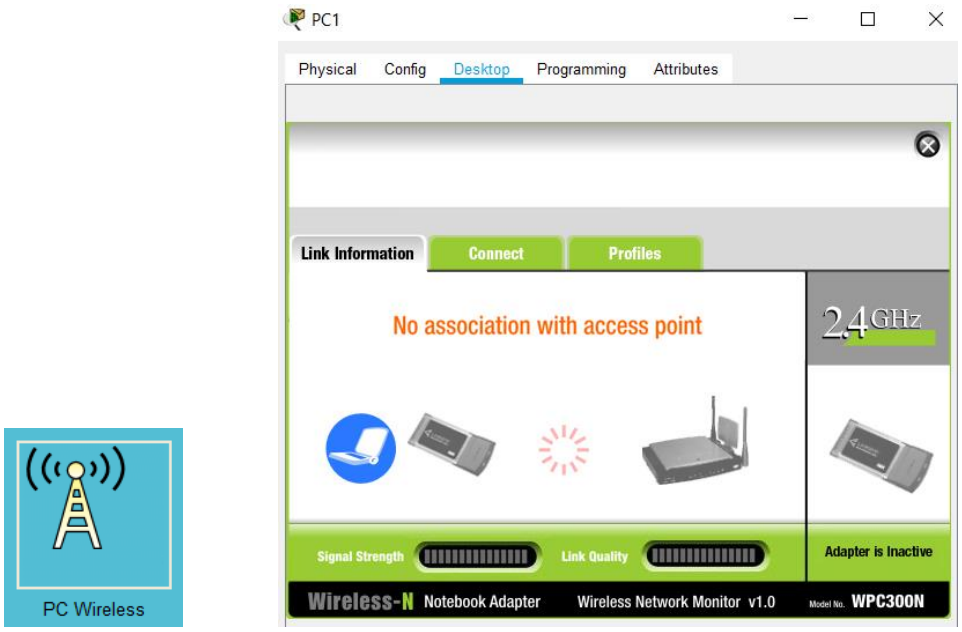


Фиг. 5.26. *Конфигурация на SSID и WEP параметри при Access Point*

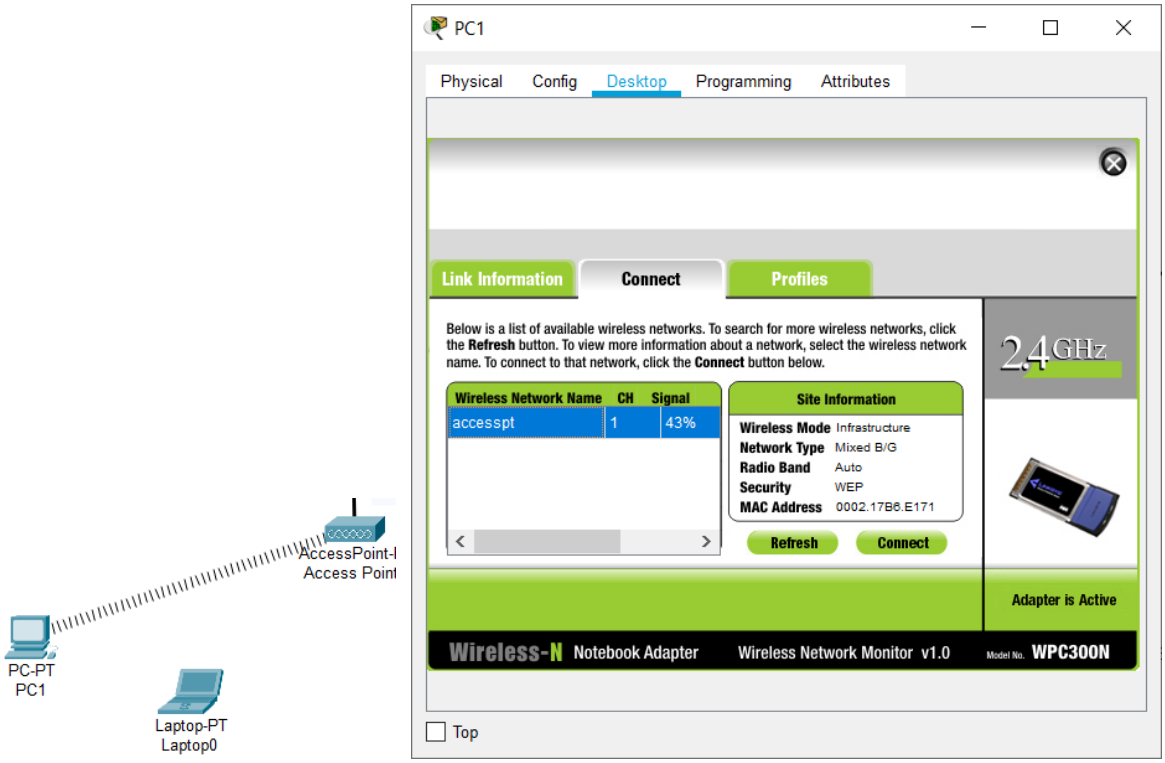


Фиг. 5.27. *Изглед на мрежовата архитектура след конфигурация на SSID и WEP параметри при Access Point*

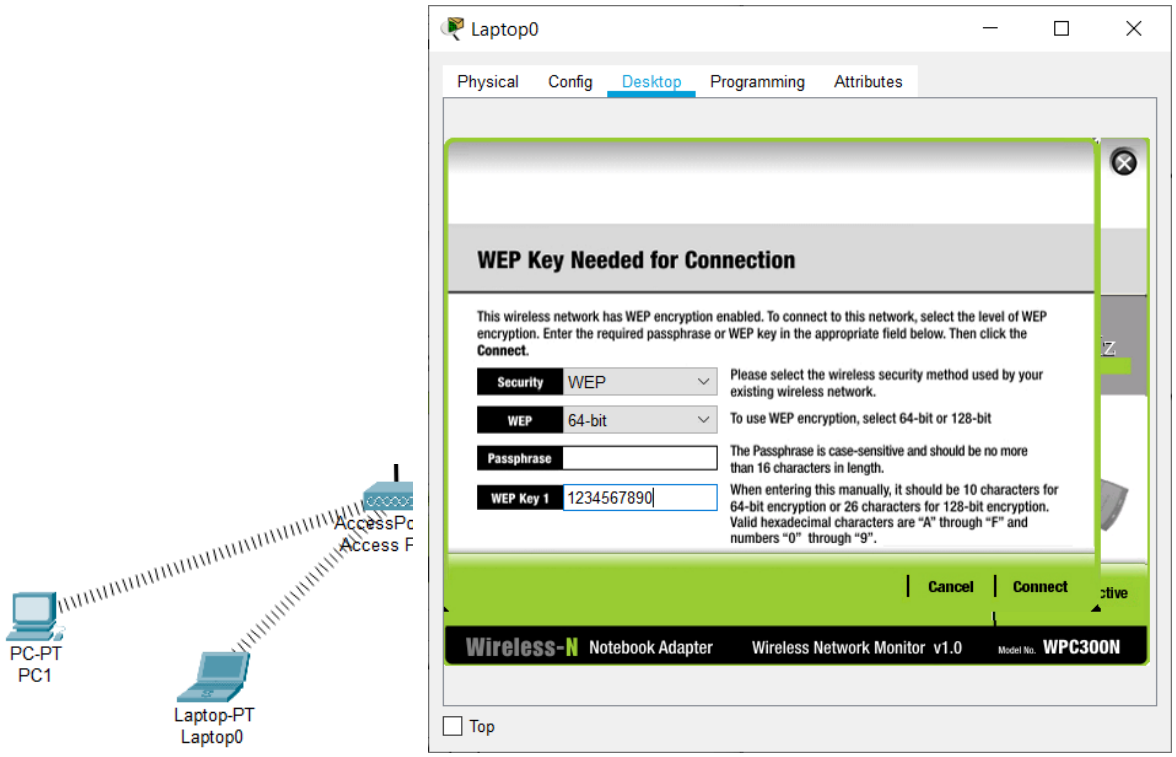
Когато конфигурацията на AP бъде завършена, както може да се забележи, всички безжични връзки от устройствата към нея са премахнати (фиг. 5.27) като трябва те наново да бъдат възстановени. Настройката на параметрите на безжичната връзката при мрежовите устройства PC1 и Laptop0 се реализира чрез прилагане за PC Wireless, показано на фиг. 5.28.



Фиг. 5.28. *Начален изглед при стартиране на приложение PC wireless при PC1*



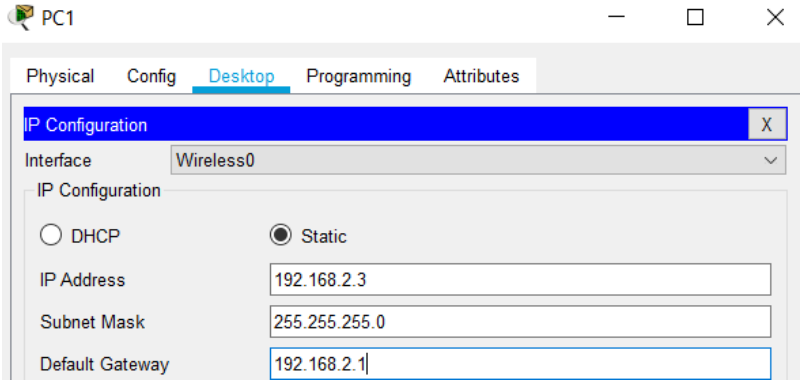
Фиг. 5.29. *Изглед при успешна активация на мрежовия адаптер при PC1*



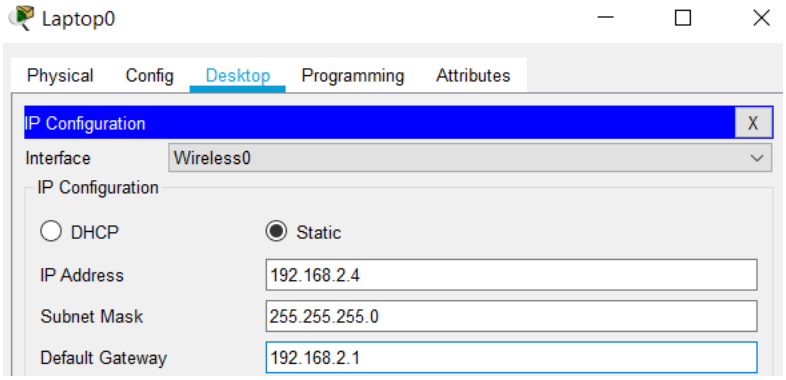
Фиг. 5.30. *Настройка на параметрите на сигурността при Laptop0*

Извършва се избор на менюто Connect и указване на мрежата, към която потребителят желае да установи връзка, в случая "accesspt" и се натиска бутона Connect. На екрана на потребителя се извежда диалогов прозорец във връзка със задаване на параметри на сигурността, където се избира режим за сигурност WEB, 64-bit тип криптиране и се въвежда криптиращия ключ WEP Key 1 "1234567890", идентичен с дефинирания при AP, и отново Connect. След което на екрана се визуализира диалогов прозорец, показващ активността на мрежовия адаптер и съответната спецификация от настройки (фиг. 5.29). Аналогично на фиг. 5.30 е даден процеса на въвеждане на параметрите за сигурност при Laptop0.

Настройките по отношение на активността на мрежовия адаптер при PC1 и Laptop0 чрез приложение PC Wireless са вече завършени и е необходима последваща IP конфигурация на устройства (IP address, Subnet Mast, Default Gateway), каквато е направена на фиг. 5.31.

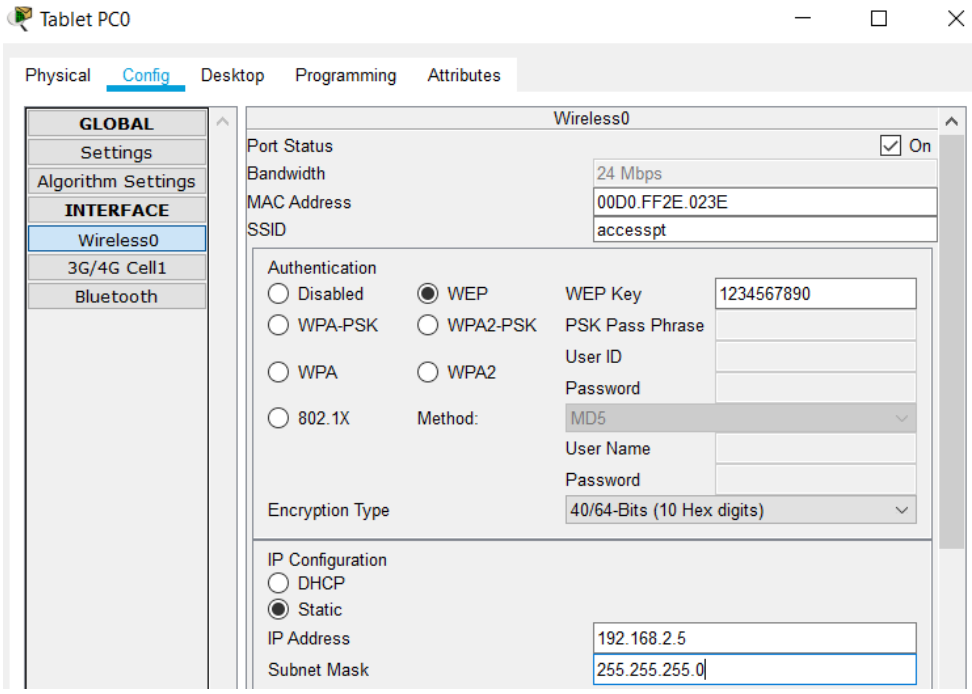


а)

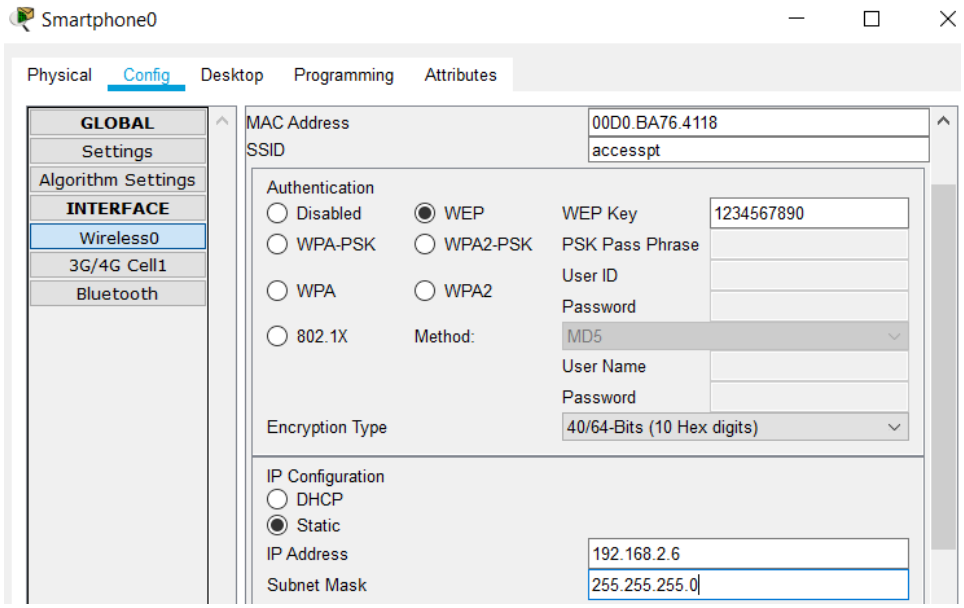


б)

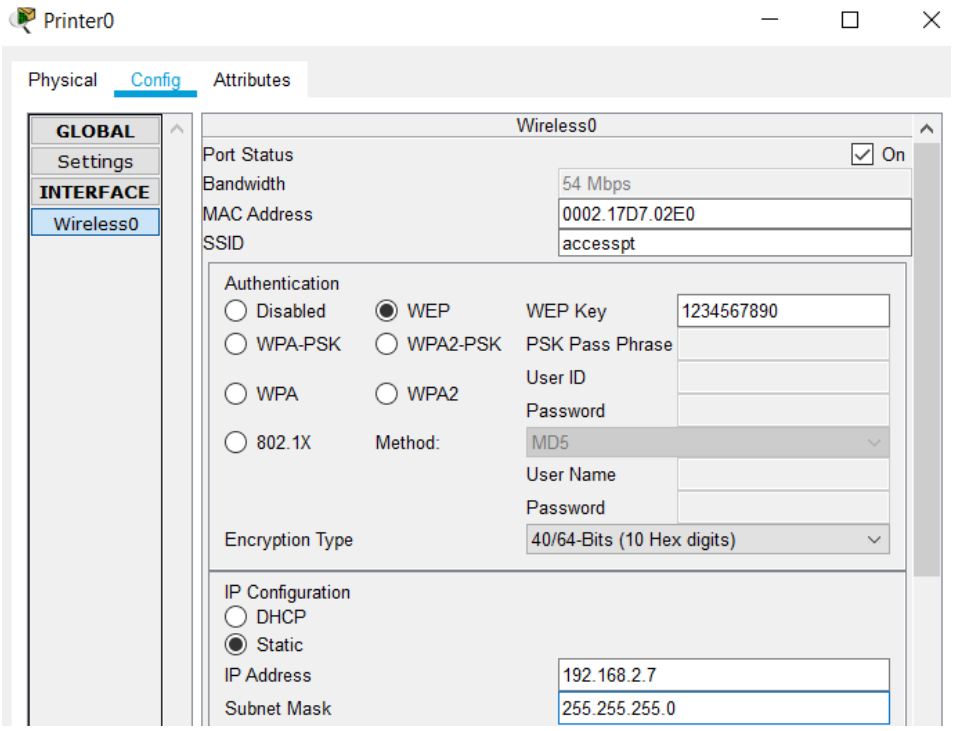
Фиг. 5.31. *IP конфигурация на а) PC1 и б) Laptop0*



а)



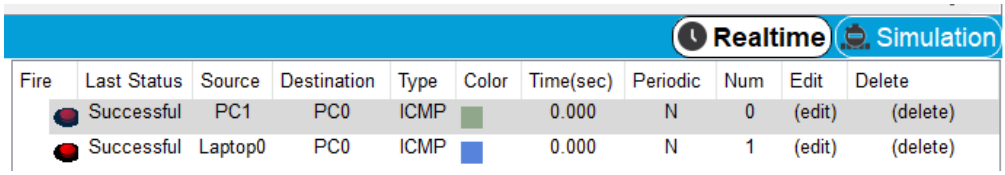
б)



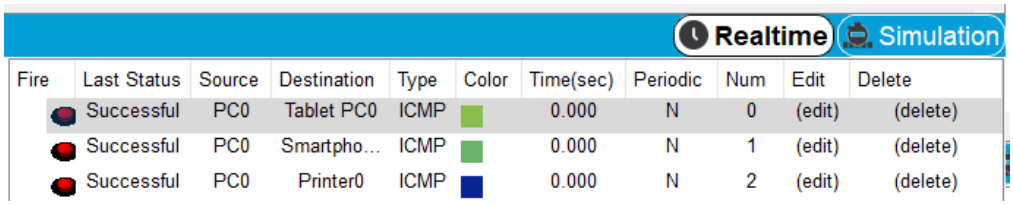
в)

Фиг. 5.32. *Конфигурация на SSID, WEP и IP параметри при а) TabletPC0, б) Smartphone0 и в) Printer0*

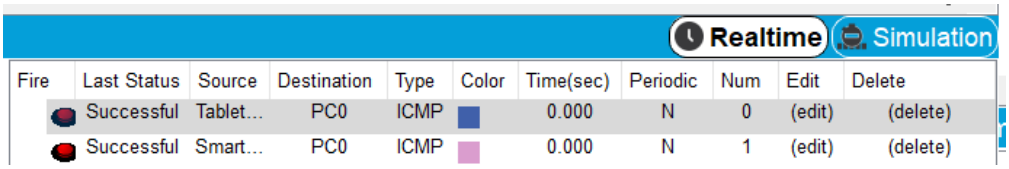
На следващ етап се извършва аналогична настройка на параметрите на сигурността – SSID идентификатор на мрежата и решим за сигурност WEP, и IP конфигурация при интерфейс Wireless0 при мрежови устройства TabletPC0, Smartphone0 и Priter0, както е представено на фиг. 5.32. Последната стъпка от изследване на функционалната мрежова архитектура с „точна се достъп“ е тестването на пакетното предаване на PDU информационни единици между мрежовите устройства, онагледено на фиг. 5.33.



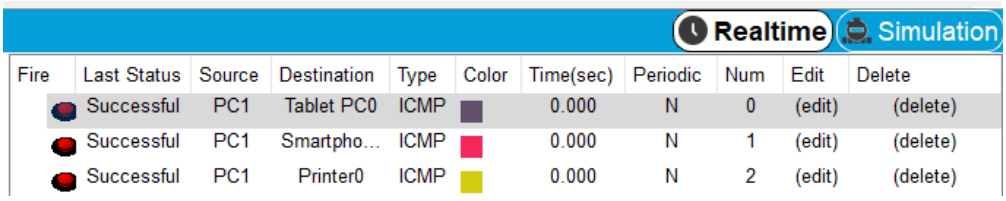
а)



б)



в)



г)

Фиг. 5.33. *Тестване чрез PDU а) "PC1 и Laptop0 към PC0", б) "PC0 към TabletPC0, Smartphone0 и Printer0", в) "TabletPC0 и Smartphone0 към PC0" и г) "PC1 към TabletPC0, Smartphone0 и Printer0"*

Тестовите процедури обхващат следните направления:

- PC1 и Laptop0 към PC0;
- PC0 към TabletPC0, Smartphone0 и Printer0;
- TabletPC0 и Smartphone0 към PC0;
- PC1 към TabletPC0, Smartphone0 и Printer0.

5.4. Контролни въпроси

1. Какви режими за администриране на оторизацията на потребители се използват при конфигурация на Wireless мрежите?
2. Каква дължина на ключовете се използват при RC4 подхода за криптиране при WEP режим за сигурност в Wireless мрежите?
3. Посочете примери за протоколи и стандарти относно контрол на достъпа и маршрутизацията при Wireless мрежови конфигурации?
4. В какво се изразява предимството на криптографски режим WPA2 пред WPA версията?
5. По какъв начин се осъществява свързаността между безжичните локални мрежи и кабелните мрежи?

Лабораторно упражнение № 6

ПРОЕКТИРАНЕ НА КОНФИГУРАЦИИ С ПАКЕТНО ПРЕДАВАНЕ ПОСРЕДСТВОМ СЕРИЙНА КОМУНИКАЦИЯ МЕЖДУ МРЕЖОВИТЕ МАРШРУТИЗАТОРИ В CISCO PACKET TRACER

6.1. Цел на упражнението

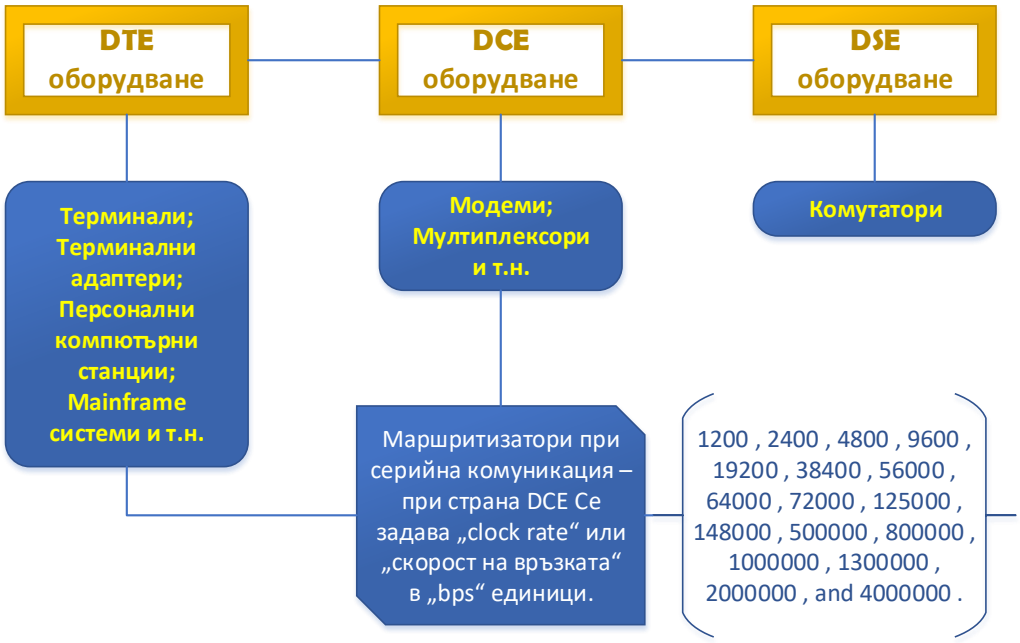
Да се запознаят студентите с различните аспекти на използването на серийна комуникация между мрежови маршрутизатори и подход за криптиране на пакетното предаване чрез използване на console lines към персоналните потребителски станции.

6.2. Теоретична постановка

Типове комуникационно оборудване в комуникационните и компютърните системи

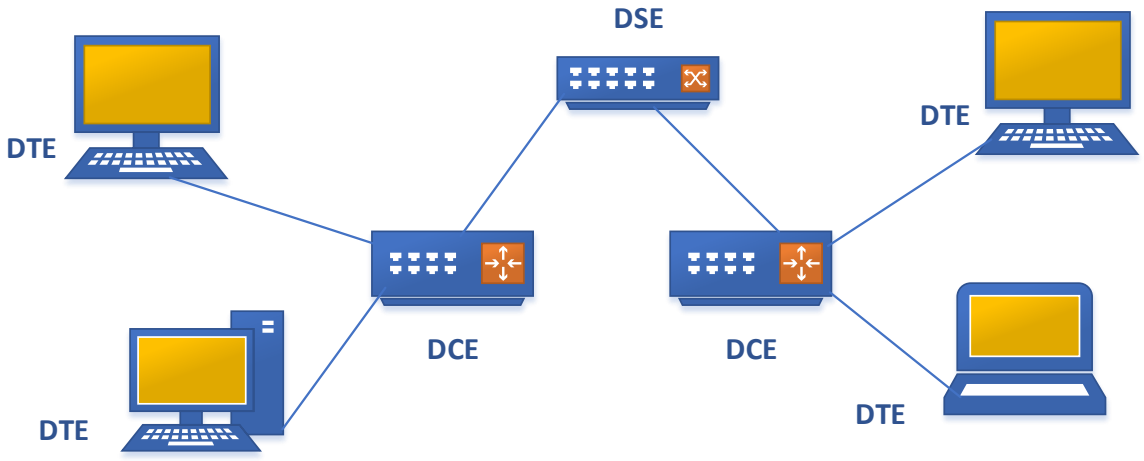
Виртуалната комуникационна система може да бъде разгледана като модел, който описва процедурен трансфер на данни между две крайни станции, наречени „предавател“ и „приемник“, по съответен комуникационен канал за връзка. Дефинират се на два основни типа системи, определящи допълнителните структурни технически средства за допълнителна обработка на предаваната информация. В състава на аналоговите комуникационни системи се включват блокове за „модулация/демодулация“ и „мултиплексиране/демултиплесиране“ на сигнали в началото и края на канала за връзка. При цифровите системи допълнително се въвеждат системни модули за „аналогово-цифрово/цифрово-аналогово преобразуване“, „кодиране и компресиране/декодиране на източника на данни“, „кодиране/декодиране на канала“ и „мултиплексиране/демултиплексиране“ [11].

От мрежова гледна точка в комуникационните и компютърните системи, използваното оборудване може да бъде разпределено в следните три категории:



Фиг. 6.1. *Категории оборудване в мрежовите комуникационни и компютърни системи*

- Data Terminal Equipment (***DTE***) – преобразуване на изходящи данни в трансмисионен сигнал и обратното;
- Data Communication Equipment (***DCE***) – позволява включване на DTE устройства. Неговото използване има отношение към установяването, поддържането и прекратяването на мрежовите сесии;
- Data Switching Equipment (***DSE***) – осигурява свързваност към DCE устройствата и следи за коректното предаване на данни в комуникационната мрежа [12].



Фиг. 6.2. *Примерна мрежова конфигурация с използването на DTE, DCE и DSE модули*

Фиг. 6.3. *Серийни DCE и DTE интерфейси в Cisco Packet Tracer*

Техническите средства, спадащи към посочените категории оборудване, са описани на фиг. 6.1, докато на фиг. 6.2 и фиг. 6.3 са дадени примерни мрежови топологии с тяхно включване и достъп до типове DCE и DTE серийни интерфейси в програмна среда Cisco Packet Tracer.

6.3. Последователност на работа и задачи за изпълнение

6.3.1. Конфигуриране на комуникация и пакетно предаване на данни между рутери по сериен интерфейс

Ще бъде разгледана възможността за конфигуриране на комуникация между мрежови сегменти, изградени посредством рутери и комутатори модели “2612XM” и “2950-24”, на основата на сериен комуникационен канал за връзка между структурните маршрутизиращи звена. Тъй като указаните рутери разполагат с Fast Ethernet тип интерфейси е необходимо да бъде включен допълнителен модул за серийна връзка. Тук

може да бъде използван модулът “WIC-1T”, чрез който към наличните “fa 0/0”, “fa 0/1” се добавя серийния порт “s0/0”, както е представено на фиг. 6.4.

Фигура 6.5 онагледява обектната мрежова архитектура, в която са обособени три мрежови зони, съответно:

- ✓ Мрежова зона №1, включваща устройства R1, S1, PC1 и PC2:

Мрежови параметри:
IP address: 192.168.1.0 Subnet mask: 255.255.255.0

Настройки на комуникационните интерфейси в мрежовия обхват:
R1 fa 0/0: 192.168.1.1

S1 fa 0/1: 192.168.1.2
S1 fa 0/2: 192.168.1.3
S1 fa 0/3: 192.168.1.4

PC1: 192.168.1.5
PC2: 192.168.1.6

- ✓ Мрежова зона №2, включваща устройства R1 и R2:

Мрежови параметри:
IP address: 192.168.2.0 Subnet mask: 255.255.255.252

Настройки на комуникационните интерфейси в мрежовия обхват:
R1 s0/0: 192.168.2.1
R2 s0/0: 192.168.2.2

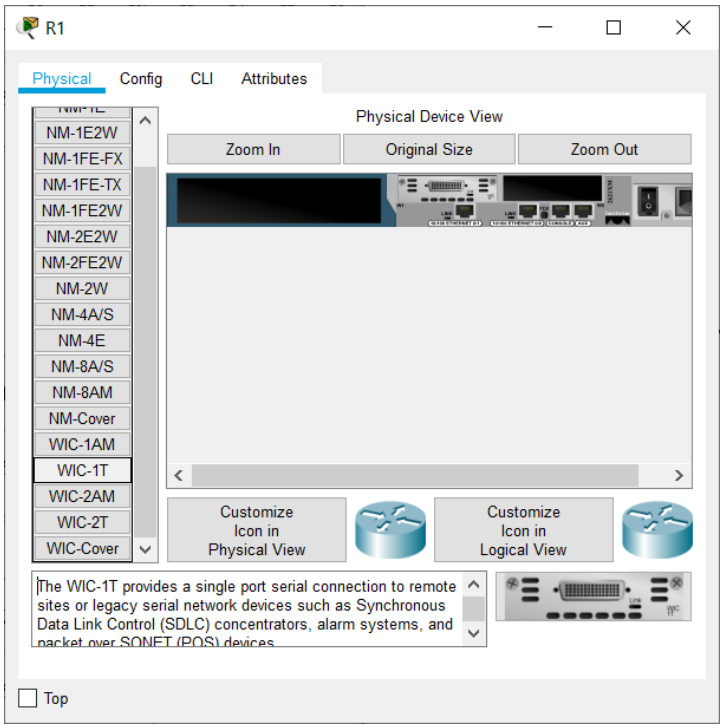
- ✓ Мрежова зона №3, включваща устройства R2, S2, PC3 и PC4:

Мрежови параметри:
IP address: 192.168.3.0 Subnet mask: 255.255.255.0

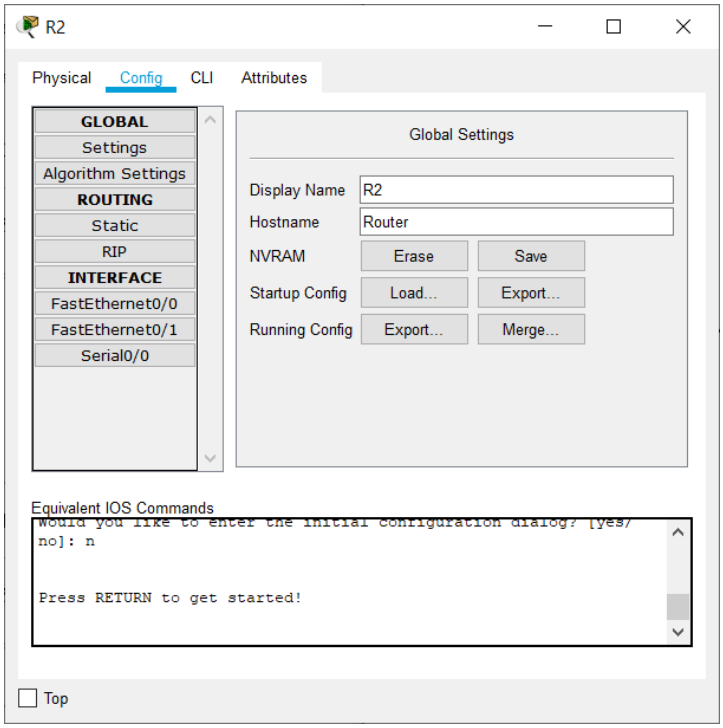
Настройки на комуникационните интерфейси в мрежовия обхват:
R2 fa 0/0: 192.168.3.1

S2 fa 0/1: 192.168.3.2
S2 fa 0/2: 192.168.3.3
S2 fa 0/3: 192.168.3.4

PC3: 192.168.3.5
PC4: 192.168.3.6

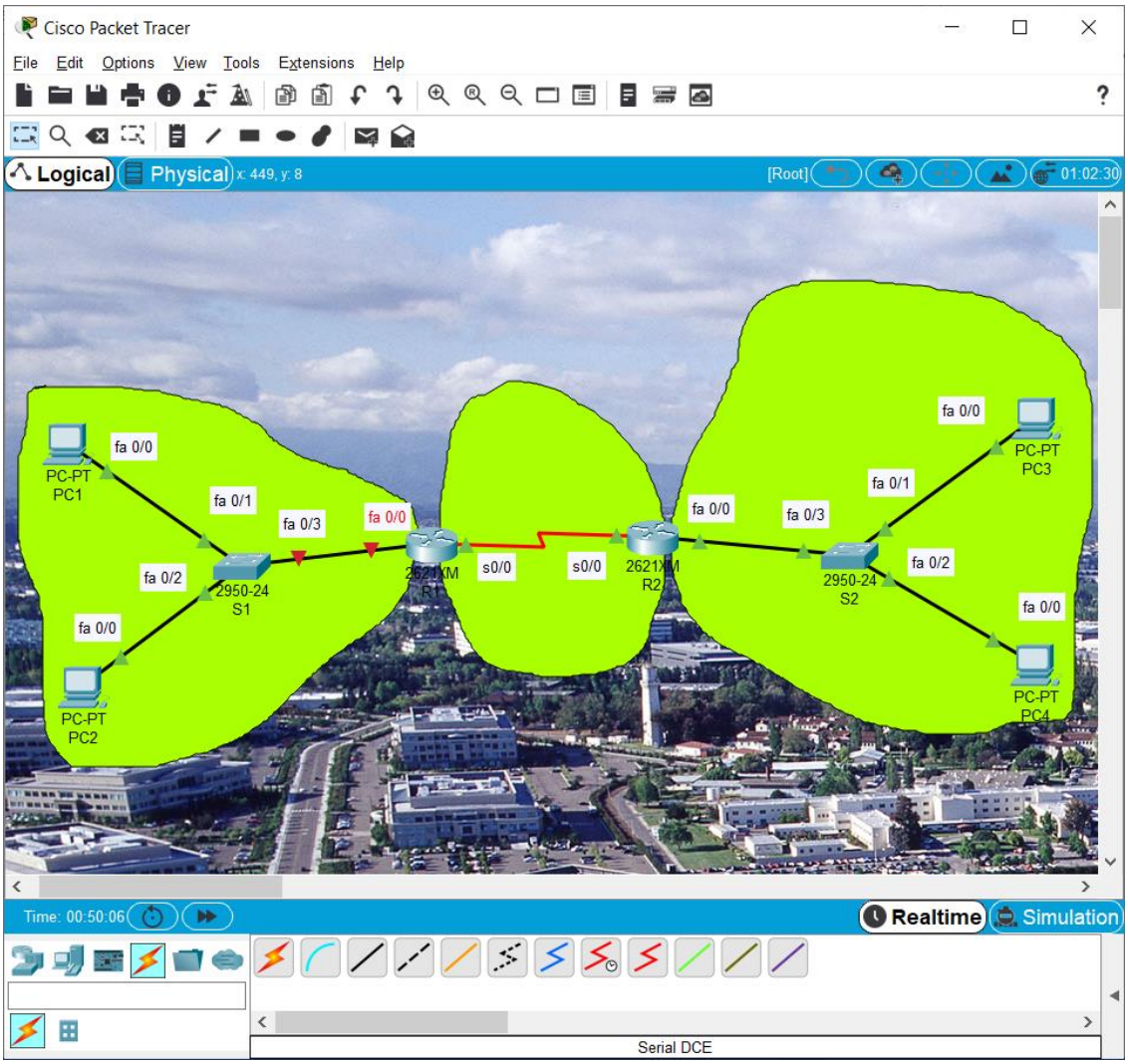


а)



б)

Фиг. 6.4. *Включване не модул за серийна комуникация WIC-1T при R1 а) и изглед на наличните портове при R2 б)*



Фиг. 6.5. *Мрежова архитектура на свързване и комуникация между два рутера посредством сериен интерфейс*

```
Router(config)#hostname R1
R1(config)#int fa 0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#no shut
R1(config-if)#exit
R1(config)#
```

Фиг. 6.6. *Настройки на интерфейс Fast Ethernet fa 0/0 при R1*

```
R1(config)#int s0/0
R1(config-if)#ip address 192.168.2.1 255.255.255.252
R1(config-if)#clock rate 128000
R1(config-if)#no shut
R1(config-if)#exit
R1(config)#
```

Фиг. 6.7. *Настройки на интерфейс Serial 0/0 при R1*

На следващ етап се извършват настройки относно устройство R1 като име на хост, IP адрес и подмрежова маска във връзка с комуникационен интерфейс “fa 0/0”, дадено

на фиг. 6.6. Също така задаване на IP адрес, Subnet mask и Crock rate (DCE (data cirquit terminal equipment) interface) по отношение на сериеният порт “s 0/0” на фиг. 6.7. По аналогичен начин се реализира конфигурация на Host name, IP address и Subnet mask, касаещи Ethernet порт “fa 0/0” за рутер R2 (фиг. 6.8), както и IP адреса и подмрежовата маска за сериения интерфейс “s 0/0” (DTE (data terminal equipment) interface) – фиг. 6.9.

```
Router>enable
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R2
R2(config)#int fa 0/0
R2(config-if)#ip address 192.168.3.1 255.255.255.0
R2(config-if)#no shut
R2(config-if)#exit
R2(config)#
```

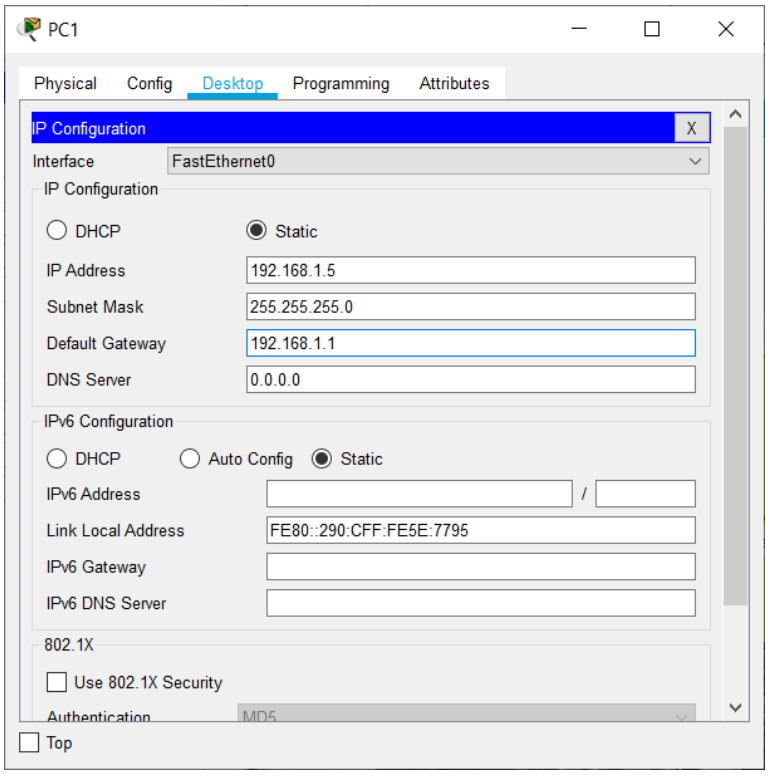
Фиг. 6.8. *Настройки на интерфейс Fast Ethernet fa 0/0 при R2*

```
R2(config)#int s 0/0
R2(config-if)#ip address 192.168.2.2 255.255.255.252
R2(config-if)#no shut
R2(config-if)#exit
R2(config)#
```

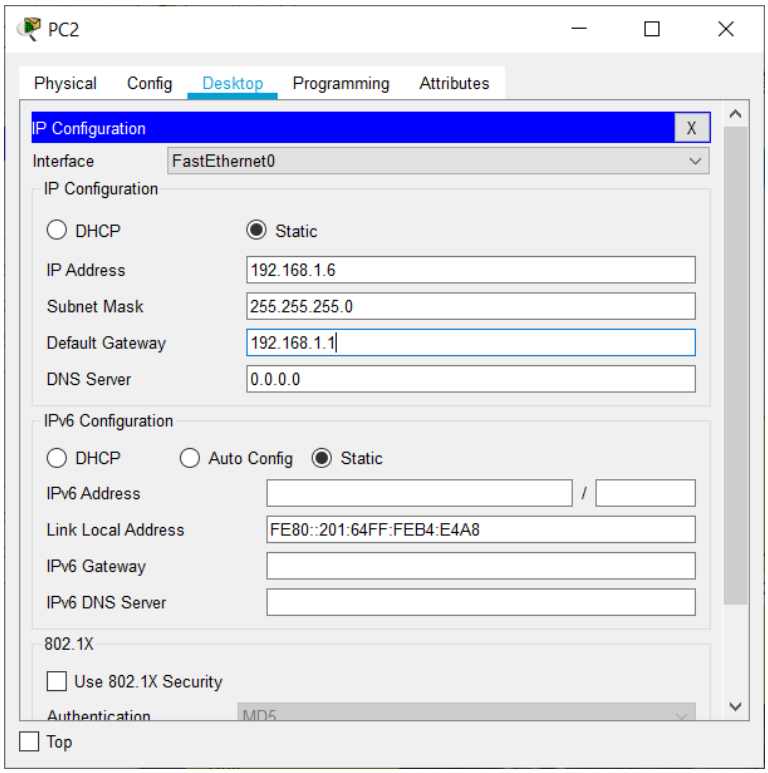
Фиг. 6.9. *Настройки на интерфейс Serial 0/0 при R2*

Друга конфигурационна стъпка се състои в дефиниране на IP адрес, Subnet mask и Default Gateway на персоналните станции от структурата на мрежовите зони №1 и №3, както следва:

- ✓ PC1 и PC2, показано на фиг. 6.10;
- ✓ PC3 и PC4, дадено на фиг. 6.11.

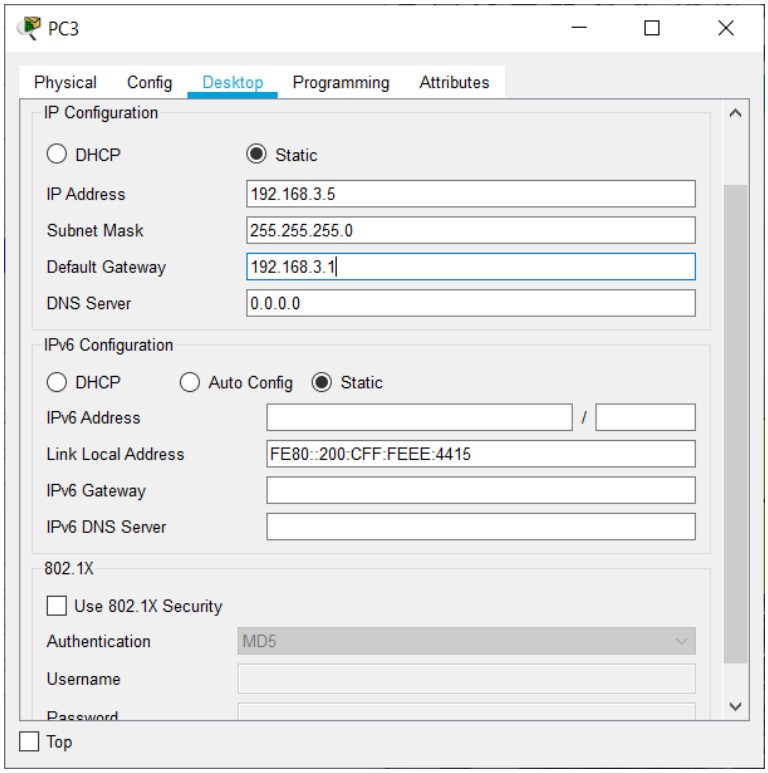


а)

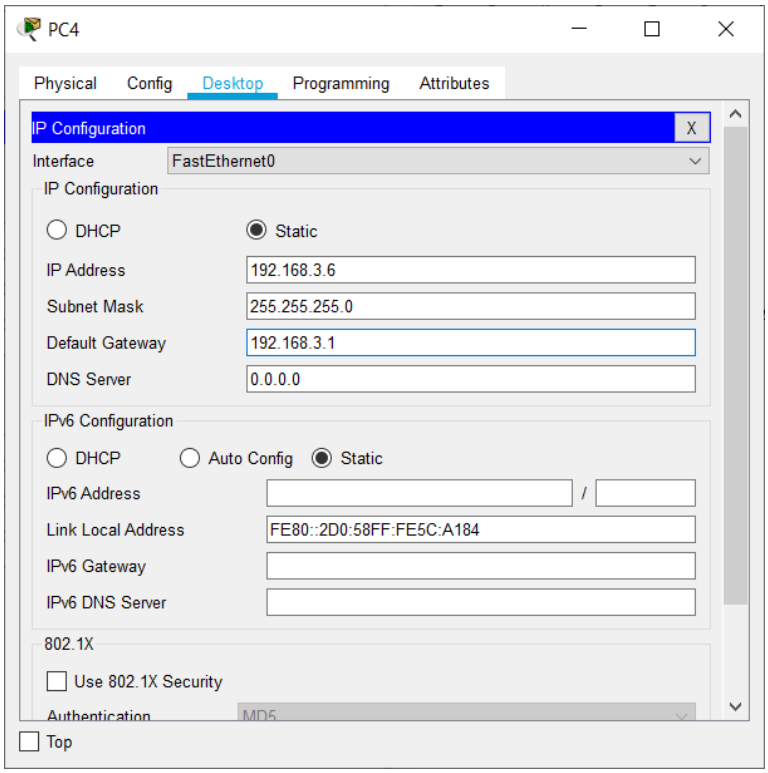


б)

Фиг. 6.10. *Конфигуриране на персонални потребителски станции а) PC1 и б) PC2*



а)



б)

Фиг. 6.11. *Конфигуриране на персонални потребителски станции а) PC3 и б) PC4*

Последните процедури по конфигурация се свеждат до установяване на статичен тип маршрутизация по информационно направление от:

- ✓ хост R1 към мрежова зона №3 – R2, по сериен интерфейс “s0/0”;
- ✓ хост R2 към мрежова зона №1 – R1, по сериен интерфейс “s0/0”.

Процедурите са последователно онагледени на фиг. 6.12.

```
R1>enable
R1#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip route 192.168.3.0 255.255.255.0 192.168.2.2
R1(config)#
```

а)

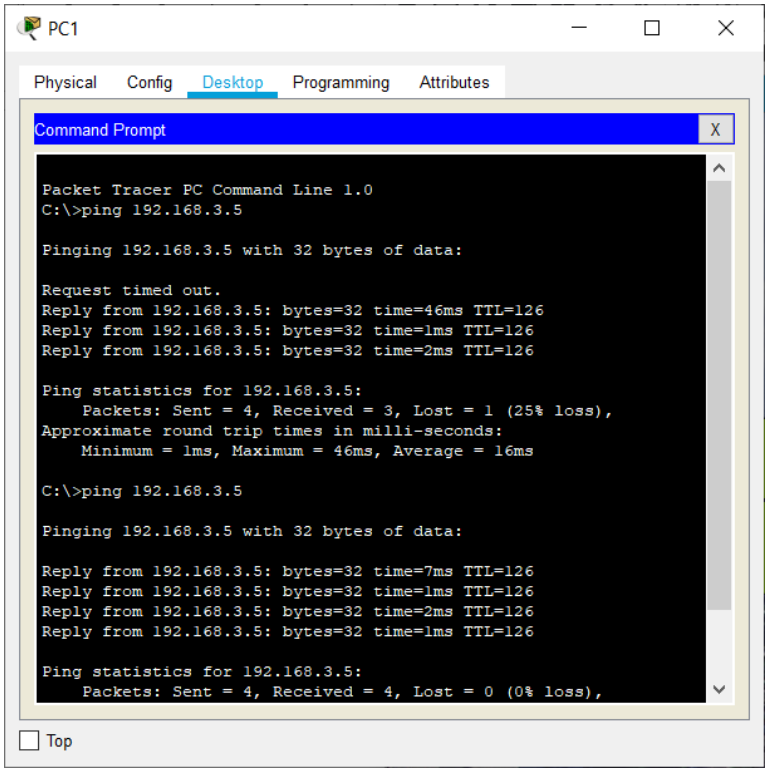
```
R2>enable
R2#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ip route 192.168.1.0 255.255.255.0 192.168.2.1
R2(config)#
R2(config)#
```

б)

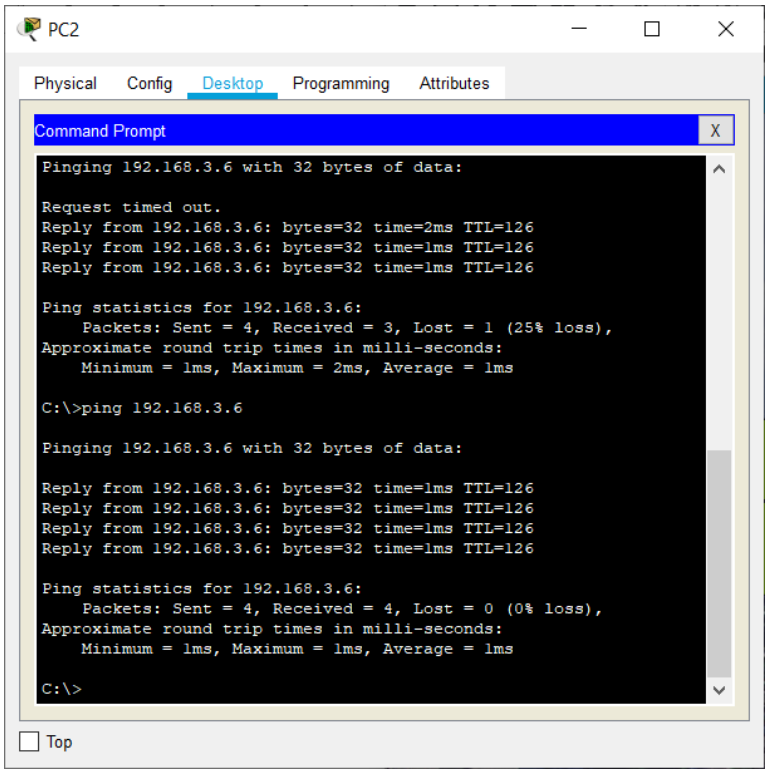
Фиг. 6.12. *Конфигуриране на статичен тип маршрутизация*
а) от мрежова зона №2 към зона №3 и б) от мрежова зона №2 към зона №1

Следва да се премине към дейности по проверка на коректността на пакетното предаване на данни чрез функционалните режими Realtime и Simulation. По отношение

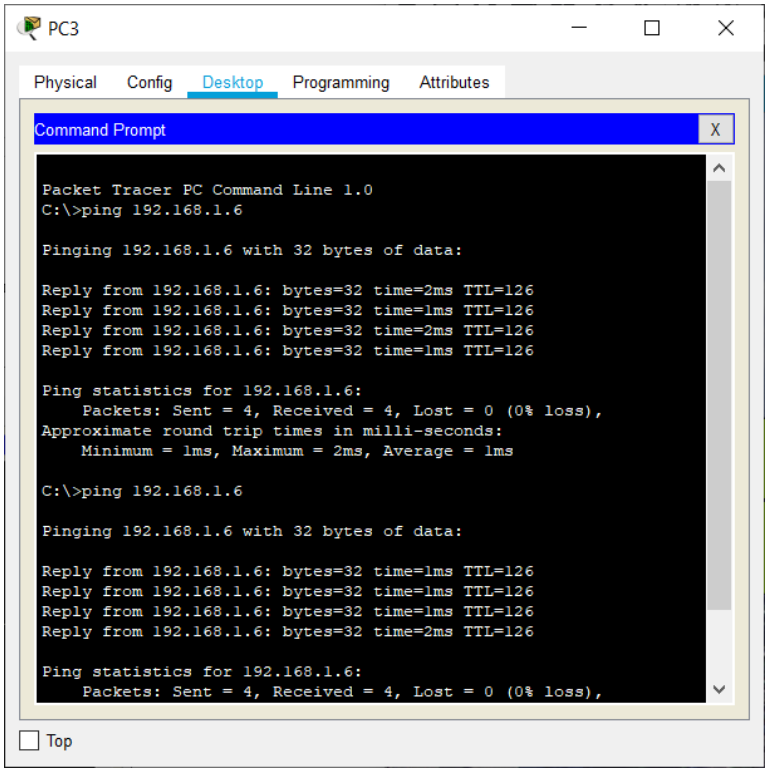
на първия режим е илюстрирана комуникацията между компютърните станции “PC1 и PC3”, “PC2 и PC4”, “PC3 и PC2” и “PC4 и PC1” на фиг. 6.13.



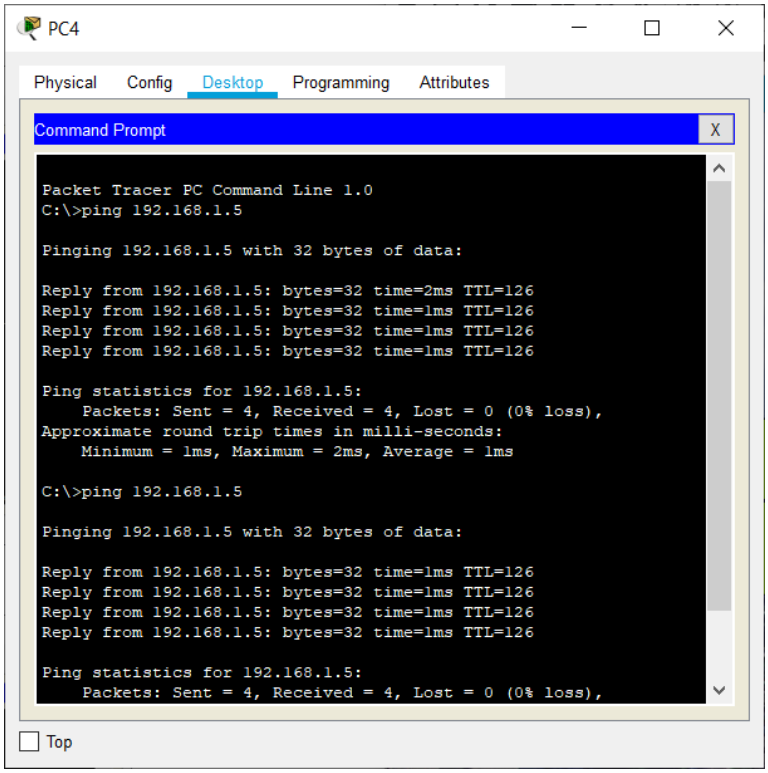
а)



б)



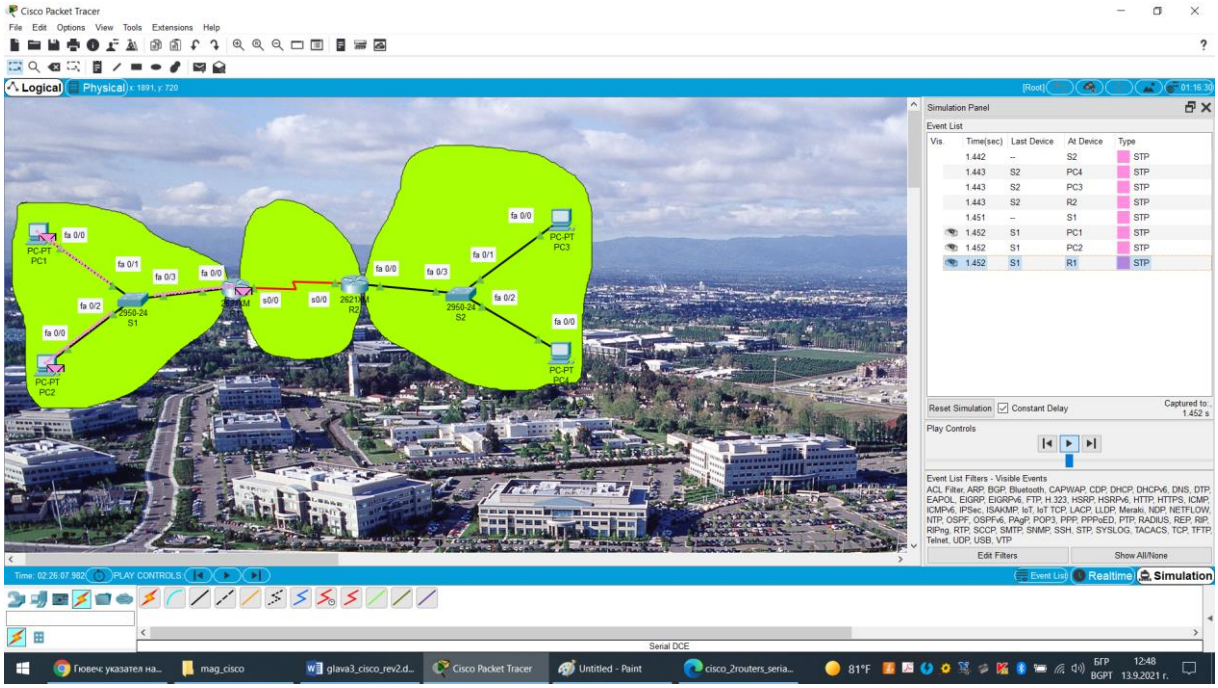
в)



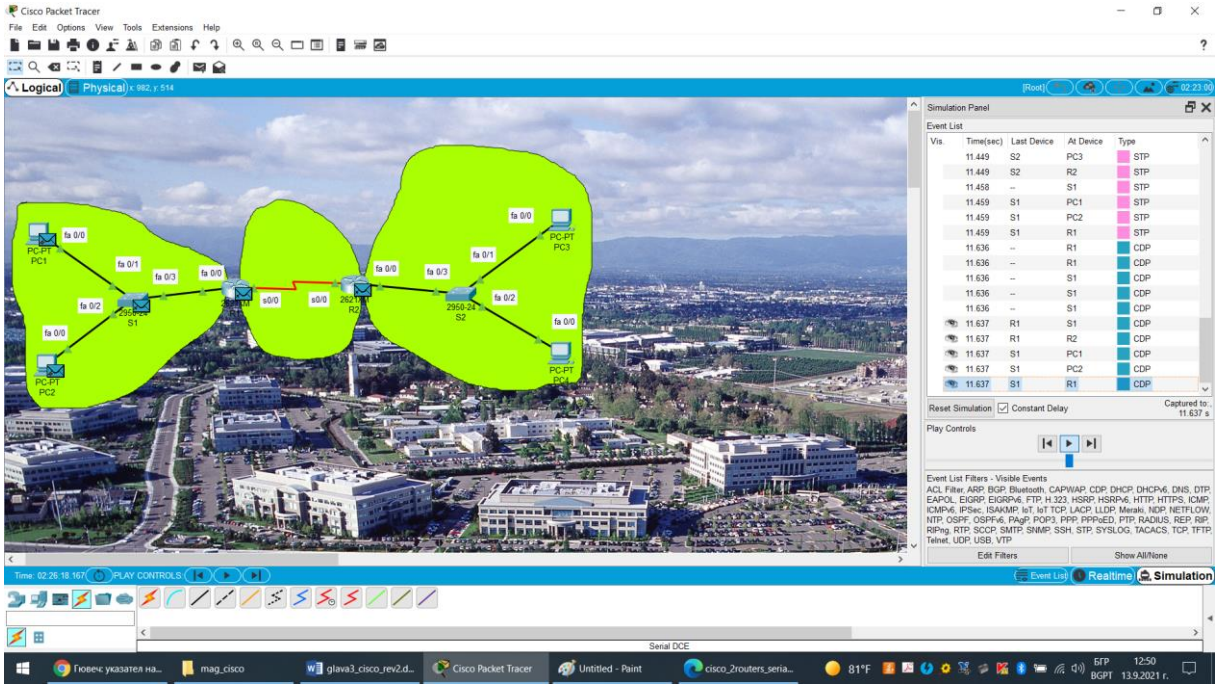
г)

Фиг. 6.13. *Предаване на данни между а) "PC1 и PC3", б) "PC2 и PC4", в) "PC3 и PC2" и г) "PC4 и PC1" в режим Realtime*

Процесите на пакетно изпращане и получаване на информационни единици могат да бъдат наблюдавани на фиг. 6.14. Може да се каже, че извършените тестови процедури по информационен трансфер между рутери по сериен интерфейс са коректни.



а)

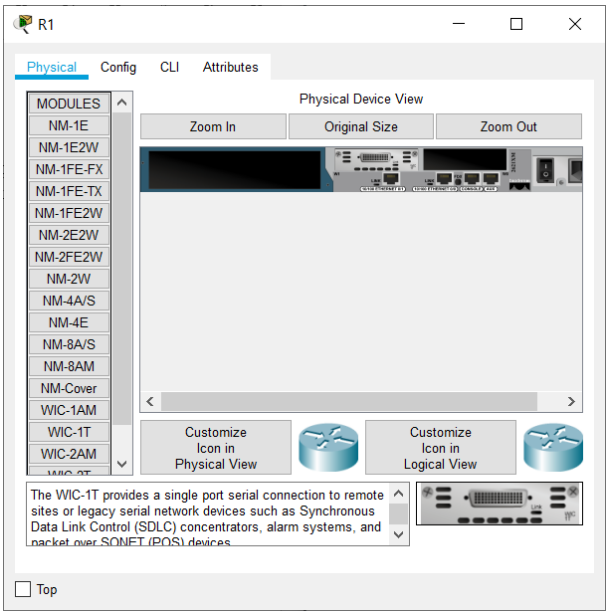


б)

Фиг. 6.14. *Предаване на данни между мрежовите устройства в симулационен режим – изглед №1 а) и изглед №2 б)*

6.3.2. Пакетно предаване на данни между хостове в конфигурация с използване на Serial и Console типове свързвания в средата на симулатор Cisco Packet Tracer.

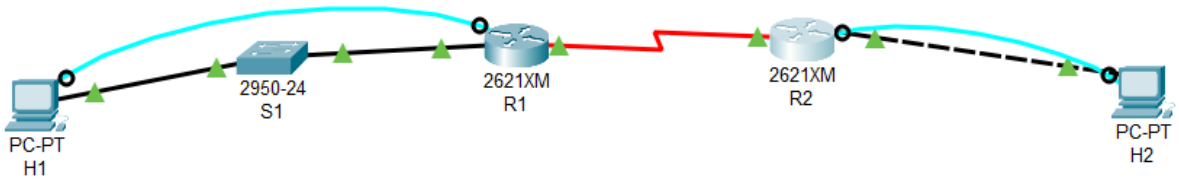
Осигуряването на информационна сигурност при пакетно предаване на данни в конфигурации между мрежови сегменти, обособени с включване на маршрутизатори при серийно свързване, може да бъде обезпечена чрез конфигуриране на *console line, виртуални портове за получаване на Telnet или SSH достъп до конкретно устройство, както и md5 Hash алгоритъм за криптиране на маршрутните връзки.* Така дефинираната процедура може да се реализира в следната последователност:



Фиг. 6.15. *Включване на модул WIC-1T за серийна комуникация относно R1 и R2*

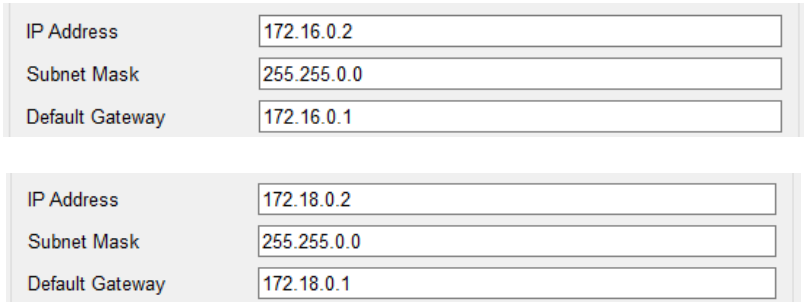
- Добавяне на модул WIC-1T при R1 и R2 за включване на опция за серийна комуникация, както е посочено на фиг. 6.15;

- Свързване на мрежовите устройства чрез меден проводник „право и обратно свързване“, сериен интерфейс и console cable. Общ вид на изследваната мрежовата архитектура е даден на фиг. 6.16:



Фиг. 6.16. *Мрежова архитектура с използване Serial и Console типове свързвания с управление на потребителския достъп на различни нива*

- Конфигуриране на хостове H1 (свързан към S1) и H2 (свързан към R2) – задаване на IP адрес, подмрежова маска и Gateway по подразбиране на фиг. 6.17;



Фиг. 6.17. *Конфигурация на хостове при мрежова архитектура с включване на Console cable и с управление на потребителския достъп на различни нива*

- Настройки на console line, vty lines (виртуални портове за получаване на Telnet или SSH достъп до конкретно устройство) и *пароли за разрешение на първо ниво на достъп до рутер R1* (User EXEC mode). Задаване на enable secret password – *криптирана посредством md5 Hash алгоритъм парола до следващо ниво на разрешение до маршрутизатор R1* (Privileged EXEC mode) – фиг. 6.18;

```
Router>enable
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R1
R1(config)#line console 0
R1(config-line)#password cisco
R1(config-line)#login
R1(config-line)#exit
R1(config)#line vty 0 4
R1(config-line)#password cisco
R1(config-line)#login
R1(config-line)#exit
R1(config)#enable secret cisco2
R1(config)#exit
```

Фиг. 6.18. *Конфигурация на console line, vty lines и md5 Hash парола при маршрутизатор R1*

- Конфигуриране на console line, vty lines (виртуални портове за получаване на Telnet или SSH достъп до конкретно устройство) и *пароли за разрешение на първо ниво на достъп до маршрутизатор R2* (User EXEC mode). Фиксиране на enable secret password – *криптирана парола чрез посредством md5 Hash алгоритъм до следващо ниво на разрешение до R2* (Privileged EXEC mode) – фиг. 6.19;

```
Router>enable
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R2
R2(config)#line console 0
R2(config-line)#password cisco
R2(config-line)#login
R2(config-line)#exit
R2(config)#line vty 0 4
R2(config-line)#password cisco
R2(config-line)#login
R2(config-line)#exit
R2(config)#enable secret cisco2
R2(config)#exit
```

Фиг. 6.19. *Конфигурация на console line, vty lines и md5 Hash парола при маршрутизатор R2*

- Дефиниране на MOTD (message-of-the-day) етикет за извеждане в случай на неоторизиран потребителски достъп и изискване за парола за достъп във връзка с маршрутизатори R1 и R2. Забраняване на маршрутизатори R1 и R2 да разрешават host names, използвайки DNS сървър. Процесът е представен на фиг. 6.20;

```
R1(config)#banner motd #Unauthorized Use Prohibited#
R1(config)#no ip domain lookup

R2(config)#banner motd #Unauthorized Use Prohibited#
R2(config)#no ip domain lookup
```

Фиг. 6.20. *Дефиниране на MOTD (message-of-the-day) при изискване за парола за достъп на маршрутизаторите R1 и R2*

- Конфигуриране на маршрутизаторите, така че console messages да не възпрепятстват command input операциите, дадено на фиг. 6.21;

```
R1(config)#line console 0
R1(config-line)#logging synchronous
R1(config-line)#exit

R2(config)#line console 0
R2(config-line)#logging synchronous
R2(config-line)#exit
```

Фиг. 6.21. *Конфигурация относно Console messages спрямо Command input операциите при маршрутизатори R1 и R2*

- Настройка на серийната комуникация – serial 0/0 interfaces, за маршрутизатори R1 и R2, показана на фиг. 6.22. Задаване на IP address, Subnet mask и Clock rate за страна DCE;

```
R1(config)#interface serial0/0
R1(config-if)#description WAN link to R2
R1(config-if)#ip address 172.17.0.1 255.255.0.0
R1(config-if)#clock rate 64000
R1(config-if)#no shutdown

R1(config-if)#
%LINK-5-CHANGED: Interface Serial0/0, changed state to up

R1(config-if)#exit

R2(config)#interface serial0/0
R2(config-if)#description WAN link to R1
R2(config-if)#ip address 172.17.0.2 255.255.0.0
R2(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/0, changed state to down
R2(config-if)#exit
```

Фиг. 6.22. *Конфигурация серийните комуникационни интерфейси на маршрутизатори R1 и R2*

- Функционалната верификация на серийната комуникация – serial 0/0 interfaces, за маршрутизатори R1 и R2 е онагледена на фиг. 6.23:

✓ R1 към R2:

```
R1#ping 172.17.0.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.17.0.2, timeout is 2
seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/8
ms
```

✓ R2 към R1:

```
R2#ping 172.17.0.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.17.0.1, timeout is 2
seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/8
ms
```

Фиг. 6.23. *Верификация на настройките относно серийните комуникационни интерфейси за маршрутизатори R1 и R2*

- Конфигуриране на FastEthernet interfaces – фиксиране на IP адрес и подмрежова маска на маршрутизатори R1 и R2, представено на фиг. 6.24;

```
R1(config)#interface fa0/0
R1(config-if)#description R1 LAN Default Gateway
R1(config-if)#ip address 172.16.0.1 255.255.0.0
R1(config-if)#no shutdown

R1(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up

R1(config-if)#exit
```

```
R2(config)#interface fa0/0
R2(config-if)#description R2 LAN Default Gateway
R2(config-if)#ip address 172.18.0.1 255.255.0.0
R2(config-if)#no shutdown

R2(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up

R2(config-if)#exit
```

Фиг. 6.24. *Конфигурация fast Ethernet интерфейсите на маршрутизатори R1 и R2*

- Съхраняване на направените конфигурационни настройки за разглежданите маршрутизатори R1 и R2, съгласно фиг. 6.25;

```
R1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

```
R2#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

Фиг. 6.25. *Съхраняване на конфигурационните настройки по отношение на маршрутизатори R1 и R2*

- Проверка на комуникацията между конфигурираните хостовете маршрутизатори, указано на фиг. 6.26 и фиг. 6.27:

```
C:\>ping 172.16.0.1

Pinging 172.16.0.1 with 32 bytes of data:

Reply from 172.16.0.1: bytes=32 time<1ms TTL=255
Reply from 172.16.0.1: bytes=32 time<1ms TTL=255
Reply from 172.16.0.1: bytes=32 time<1ms TTL=255
Reply from 172.16.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 172.16.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Фиг. 6.26. *Верификация на пакетното предаване на данни между H1 към R1*

```
C:\>ping 172.18.0.1

Pinging 172.18.0.1 with 32 bytes of data:

Reply from 172.18.0.1: bytes=32 time=1ms TTL=255
Reply from 172.18.0.1: bytes=32 time<1ms TTL=255
Reply from 172.18.0.1: bytes=32 time<1ms TTL=255
Reply from 172.18.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 172.18.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

Фиг. 6.27. *Верификация на пакетното предаване на данни между H2 към R2*

- Тестване на комуникацията между мрежовите хостове (некоректността на предаване на пакетни се обуславя от необходимостта от конфигуриране на default routers или използване на протоколи за динамично маршрутизиране). Процедурите са посочени на фиг. 6.28 и фиг. 6.29:
  - ✓ H1 към H2:

```
C:\>ping 172.18.0.2

Pinging 172.18.0.2 with 32 bytes of data:

Reply from 172.16.0.1: Destination host unreachable.
Reply from 172.16.0.1: Destination host unreachable.
Reply from 172.16.0.1: Destination host unreachable.
Reply from 172.16.0.1: Destination host unreachable.

Ping statistics for 172.18.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Фиг. 6.28. *Тестване на коректността на предаването на данни между H1 и H2*

✓ H2 към H1:

```
C:\>ping 172.16.0.2

Pinging 172.16.0.2 with 32 bytes of data:

Reply from 172.18.0.1: Destination host unreachable.
Request timed out.
Reply from 172.18.0.1: Destination host unreachable.
Request timed out.

Ping statistics for 172.16.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Фиг. 6.29. *Тестване на коректността на предаването на данни между H2 и H1*

- Конфигуриране на default routers отоносно R1 и R2 чрез използване на exit-interface mode *при показана текуща оторизация чрез въвеждане на паролите за потребителски достъп на съответни нива на сигурност* – фиг. 6.30;

```
Password:

R1>enable
Password:
R1#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip route 0.0.0.0 0.0.0.0 serial0/0
R1(config)#exit
```

а)

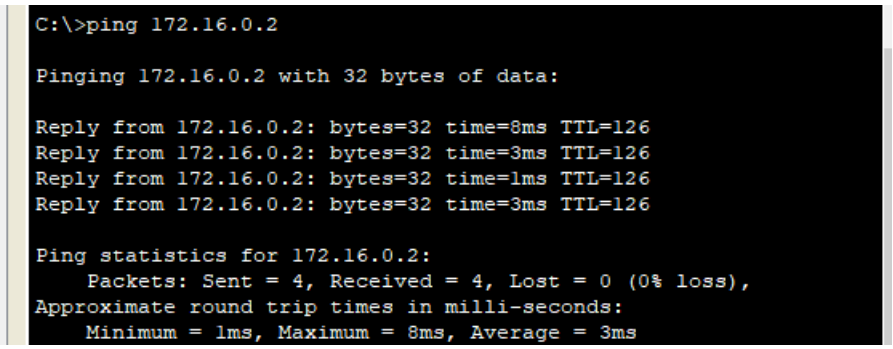
```
Unauthorized Use Prohibited

User Access Verification

Password:

R2>enable
Password:
R2#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ip route 0.0.0.0 0.0.0.0 serial0/0
R2(config)#exit
```

б)

Фиг. 6.30. *Конфигуриране на default routers по отношение на а) R1 и б) R2*

```
C:\>ping 172.18.0.2

Pinging 172.18.0.2 with 32 bytes of data:

Reply from 172.18.0.2: bytes=32 time=7ms TTL=126
Reply from 172.18.0.2: bytes=32 time=1ms TTL=126
Reply from 172.18.0.2: bytes=32 time=1ms TTL=126
Reply from 172.18.0.2: bytes=32 time=3ms TTL=126

Ping statistics for 172.18.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 7ms, Average = 3ms
```

Фиг. 6.31. *Повторно тестване на коректността на пакетно предаване на данни между хостове H1 и H2*



Фиг. 6.32. *Повторно тестване на коректността*
на пакетно предаване на данни между хостове H2 и H1

- Повторно тестване на комуникацията във връзка с коректността на предаването на пакетни данни между крайните обособени възли - мрежовите хостове, онагледено на фиг. 6.31 и фиг. 6.32.

6.4. Контролни въпроси.

1. Какви технически процеси се изпълняват в системите за комуникации при предаването на информация?
2. В какви групи може да бъде категоризирано техническото оборудване в комуникационните и компютърните системи?
3. Какъв алгоритъм за криптиране по отношение на маршрутните връзки се въвежда чрез използване на console lines?
4. При какъв режим и ниво на сигурност се извършва настройка на специфицираните console line и vty lines в мрежовите маршрутизатори?
5. В кой режим се прилага md5 Hash алгоритъмът относно регламентирането на сигурността при мрежовите маршрутизатори?

Лабораторно упражнение № 7

АНАЛИЗ И МОДЕЛИРАНЕ НА ПАРАМЕТРИЧНИ ПРОЦЕСИ И МРЕЖОВИ СЕГМЕНТИ В КОНФИГУРАЦИИ С ОПТИЧНА ПРЕНОСНА СРЕДА С ПОМОЩТА НА MATLAB И CISCO PACKET TRACER

7.1. Цел на упражнението

Да се запознаят студентите с възможостите за изследване и оценка на предаваните сигнали в комуникационните канали при специфичен тип модулация и филтрация с помощта на „Диаграми на разсейване“ и „Око-диграми“ в MATLAB, както и моделиране на мрежови сегменти с интеграция на оптична преносна среда в Cisco Packet Tracer.

7.2. Теоретична постановка

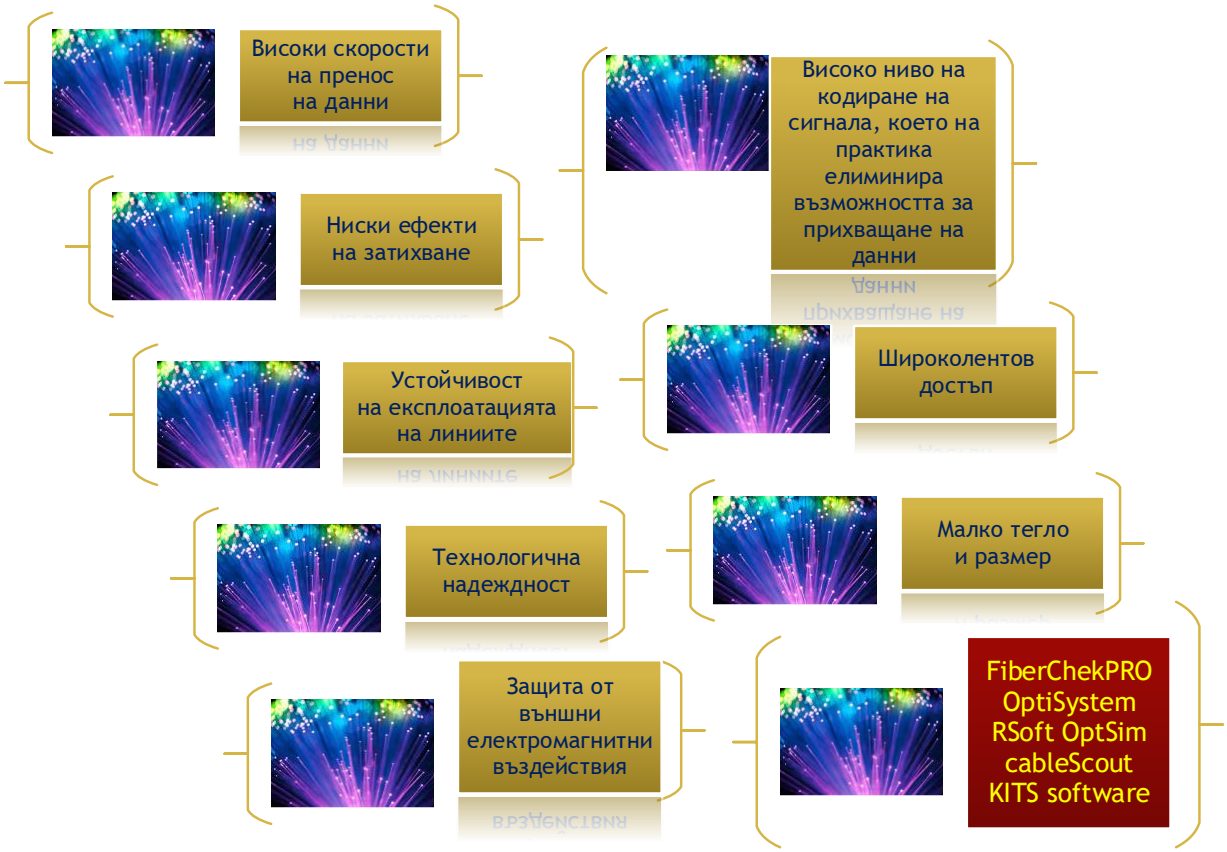
Според различни източници нашата страна е сред най-напредналите държави-членки на ЕС по отношение на скоростта на предоставяния широколентов достъп – над 98,1 % от линиите са със скорост над 2 Mbps, а над 74.1 % са със скорост между 10 и 30 Mbps, което ги прави готови за бъдещите интернет приложения, каквито са например телевизията с висока разделителна способност и висококачествени видео разговори. Това до голяма степен се дължи на факта, че делът на наследените остарели инфраструктури е твърде малък и голяма част от ново изградените фиксирани широколентови линии са по технологията FTTx на базата на оптични влакна. Това създава добра основа за предстоящото широко разгръщане на мрежите за достъп от следващо поколение (NGA - Next Generation Access). В тази връзка Национален план за инфраструктура за достъп от следващо поколение (NGN) предвижда [13]:

- Инвестиционни проекти за изграждане на широколентов достъп в отдалечените, слабонаселените и селските райони, финансирани чрез новите оперативни програми;
- Технологично обновяване и надграждане на съществуващите стари широколентови мрежи с цел постигане на необходимите параметрите, осигуряващи възможност за предоставяне на нови и модерни електронни услуги със значително по-високи скорости;
- Поетапно изграждане на мрежи за широколентов достъп от ново поколение (NGA).

Разработчиците и инженерните специалисти, създаващи и внедряващи проекти за осигуряване на широколентови услуги на базата на оптични преносни трасета, трябва да притежават специализирани познания и умения относно хардуерните ресурси и измерваните параметри за качество на пренасяната информация. Основните положителни аспекти на работата на оптичните линии са свързани с високата скорост на предаване на информацията. Доскоро тази стойност беше изразена като рекордна цифра. Сега обаче и тези данни се считат за неподходящи по отношение на рекордните цифри.

Така новата технология на вълновите мултиплексиращи системи позволява оптични влакна да осигуряват скорост на сигнала от 15 Tbps. Големите телекомуникационни корпорации практикуват използването на многоканални оптично-оптични комуникации

на разстояние до 10000 км със скорост 100 Gb /s. Могат да бъдат обобщени следните предимства на оптичните технологии, както и софтуерните инструменти за анализ на оптични мрежи, посочени на фиг. 7.1 [13, 14].



Фиг. 7.1. *Предимства на оптичните технологии и инструменти за анализ на параметрите на оптични мрежи*

7.3. Последователност на работа и задачи за изпълнение.

7.3.1. Генериране, визуализация и изследване на комуникационни сигнали чрез „диаграми на разсейване“ и „око-диаграми“ в MATLAB среда.

Използването на подходящи графични инструменти за изследване на сигнали в комуникационните системи на различни фази от тяхната обработка при предаване е важна задача относно осигуряването на качеството на информацията. Едни от най-подходящите инструменти за това са ***Scatter Plots*** и ***Eye Diagrams*** – „диаграми на разсейване“ и „око-диаграми“.

За целта е използван ***QPSK (Quadrature Phase Shift Keying) модулиран символен сигнал, върху който е приложен square-root и raised cosine (RRC) филтър***. Прилагането на основни типове сигнални обработки протичат съгласно следната последователност:

- ***Прилагане на „Диаграми на разсейване“ – Scatter plots***
  - Настройване на параметрите на RRC филтъра – Filter span, Rolloff factor и Samples per symbol. RRC филтриращото звено е известно още като „филтър с повишен косинус“. Често се прилага при манипулиране на формата на импулсите в цифровата модулация, поради свойството за минимизация на междусимволните смущения.

```
span = 10;

rolloff = 0.2;

sps = 8;
```

- Получаване на коефициентите на RRC филтъра чрез функцията rcosdesign на база на въведените конфигурационни параметри:

```
filtCoeff = rcosdesign(rolloff, span, sps);
```

- Генериране на случайна символна поредица относно фиксирана големина на символите „4“:

```
rng('default')
data = randi([0 3],5000,1);
```

- Прилагане на Quadrature Phase Shift Keying тип модулация върху генерираната случайна поредица от символи:

```
dataMod = pskmod(data, 4, pi/4);
```

- Използване на филтация върху модулираната случайна символна поредица:

```
txSig = upfirdn(dataMod,filtCoeff, sps);
```

- Калкулация на SNR параметъра (Signal-to-Noise Ratio) по отношение на семплирания QPSK модулиран сигнал:

```
EbNo = 20;
snr = EbNo + 10*log10(2) - 10*log10(sps);
```

- Налагване на AWGN (Additive White Gaussian Noise) въздействие върху *предавания* сигнал:

```
rxSig = awgn(txSig,snr,'measured');
```

- Прилагане на Root Raised Cosine филтъра относно вече *получения* сигнал:

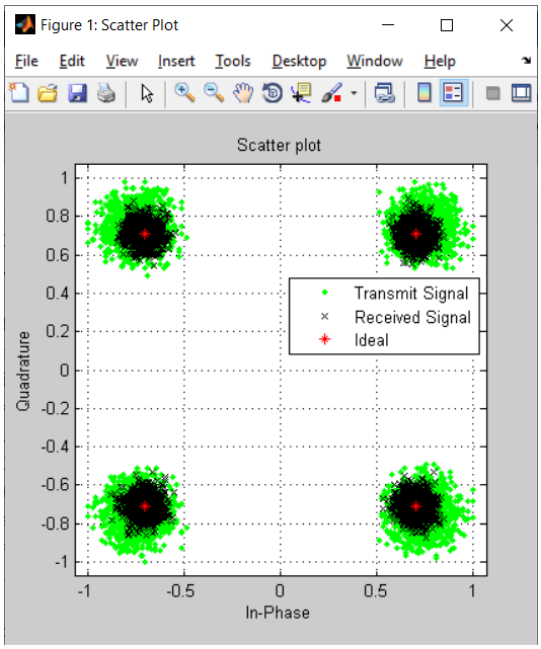
```
rxSigFilt = upfirdn(rxSig, filtCoeff, 1, sps);
```

- Извършване на демодулация по отношение на филтрирания сиглал на основата на RRC филтъра:

```
dataOut = pskdemod(rxSigFilt, 4, pi/4, 'gray');
```

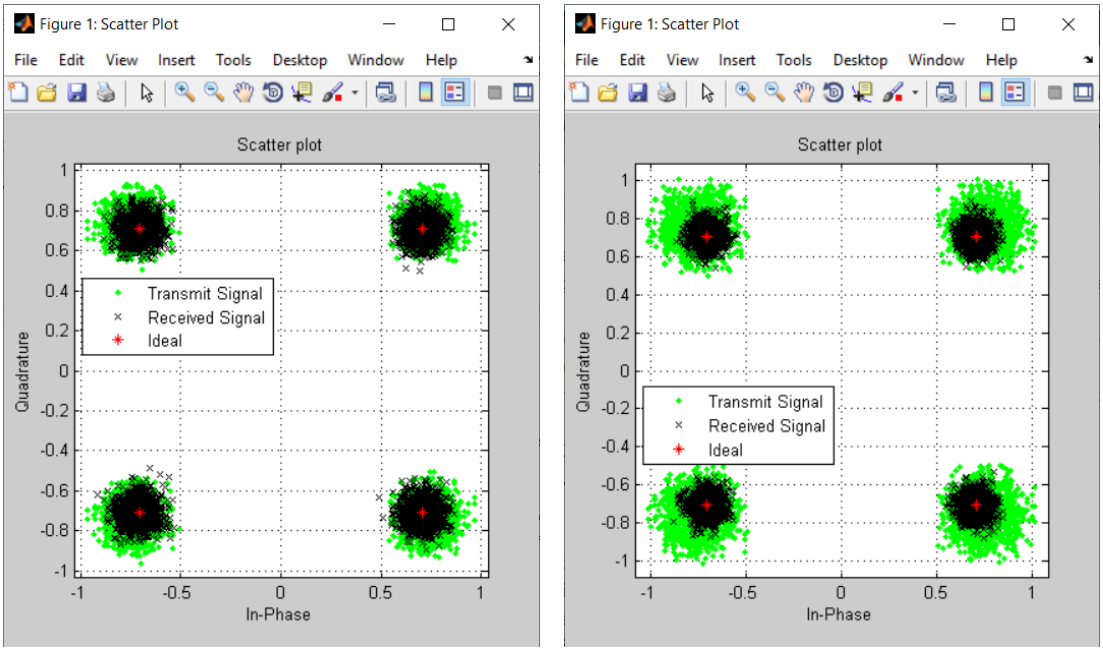
- Използване на „диаграми на разсейване“ за визулизация на филтрираната QPSK модулирана символна поредица преди добавяне на AWGN - ‘Transmit signal’, и филтрирания през RRC звено QPSK модулиран сигнал с насложен AWGN - ‘Receive signal’, както и QPSK модулираната информационна поредица непосредствено след генериране - ‘Ideal’. Процесите са онагледени на фиг. 7.2:

```
h = scatterplot(sqrt(sps) * txSig(sps*span+1:end-sps*span), sps, 0, 'g.');
hold on; scatterplot(rxSigFilt(span+1:end-span), 1, 0, 'kx', h)
scatterplot(dataMod, 1, 0, 'r*', h)
legend('Transmit Signal','Received Signal','Ideal','location','best'); grid;
```



Фиг. 7.2. *Диграма на разсейване на модулиран QPSK и получен RRC филтриран сигнал при параметри на филтъра - span = 10; rolloff = 0.2; sps = 8*

- Повторение на съставената процедура с изследване на ефекта от промяна на филтърните RRC параметри, както е показано на фиг. 7.3:

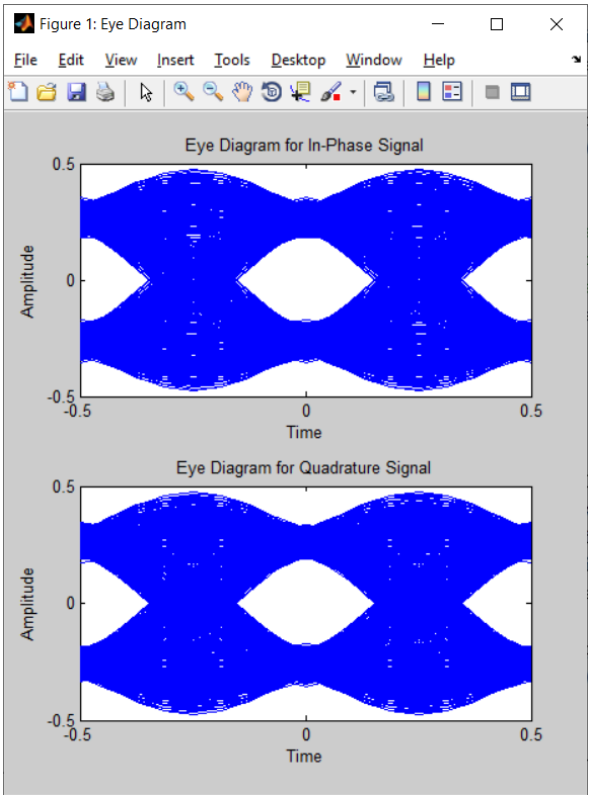


а) *б)*

Фиг. 7.3. *Диграма на разсейване на модулиран QPSK и получен RRC филтриран сигнал при параметри на филтъра - span = 10, sps = 8 и а) rolloff = 0.15 и б) rolloff = 0.25*

❖ ***Прилагане на „Око-диаграми“ – Eye diagrams***

- Генериране на Eye diagram за два символни периода на филтрирания модулиран сигнал (получен непосредствено преди пресмятане на SNR критерия) на фиг. 7.4:



Фиг. 7.4. *Око-диаграма на филтрирания модулиран сигнал за два символни периода*

- Промяна на стойността на *rolloff* фактора на RRC филтъра за визуализация на ефекта от Око-диаграмата:

```
rolloff = 0.3;
```

- Генериране на новите филтърни RRC коефицинти на базата на повишения *rolloff* фактор:

```
filtCoeff = rcosdesign(rolloff, span, sps);
```

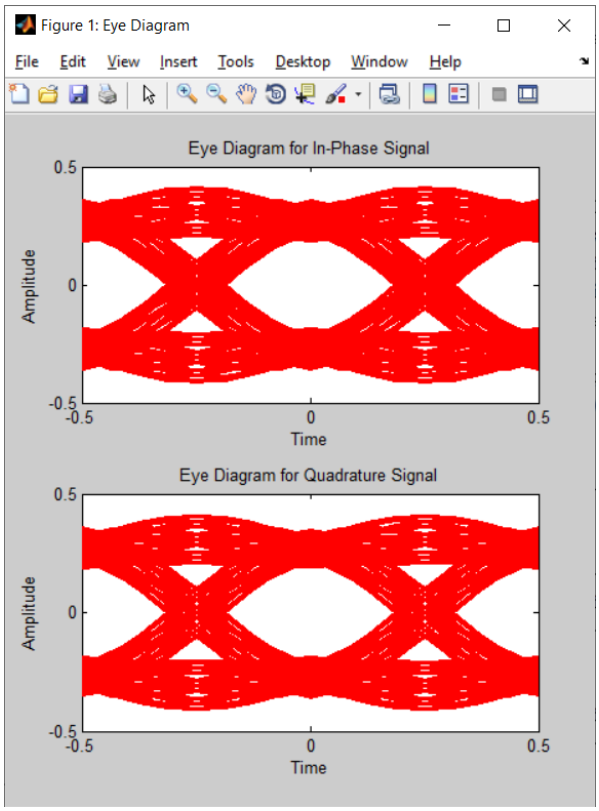
- Прилагане на RRC филтриращото звено спрямо QPSK модулираната символна поредица (сигналът непосредствено след генериране):

```
txSig = upfirdn(dataMod,filtCoeff, sps);
```

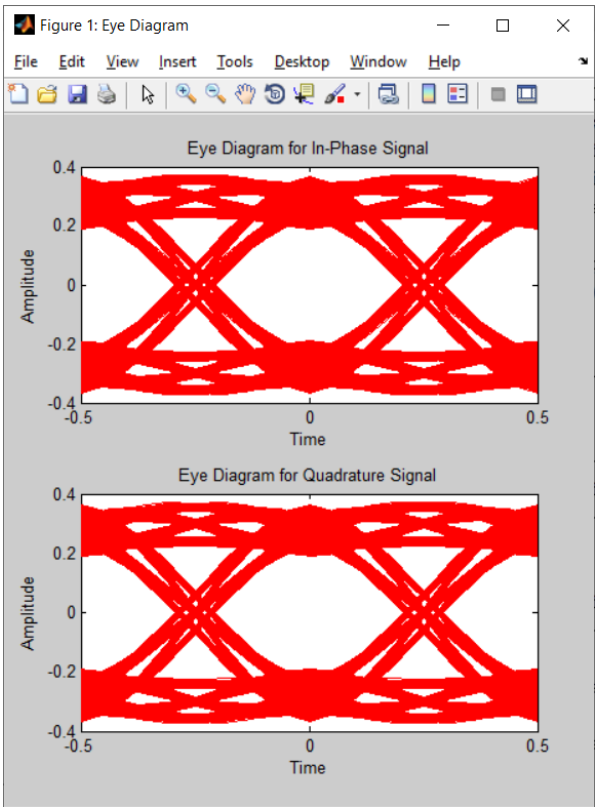
- Генериране на Eye diagram за два символни периода на новия филтриран модулиран сигнал (сигналът непосредствено преди пресмятане на SNR критерия и добавяне на AWGN въздействие):

```
eyediagram(txSig(sps*span+1:end-sps*span), 2*sps, 1, 0, 'r')
```

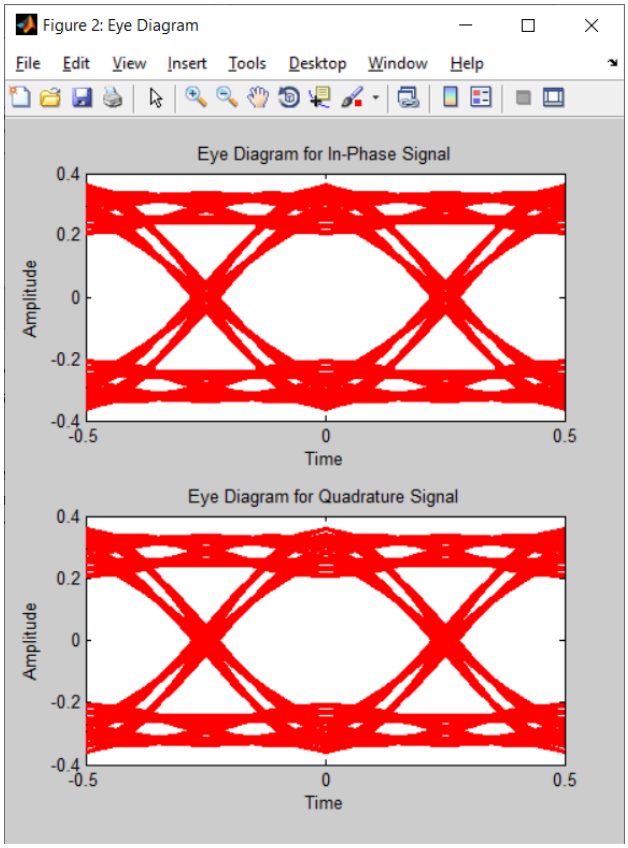
- Повторение на стъпките за създаване и използване на RRC филтъра при повишаване на *rolloff* фактора на 0.4 и 0.5, както и визуализация на изходните Око-диаграми относно изследването на ефекта, представено на фиг. 7.5:



а)



б)



в)

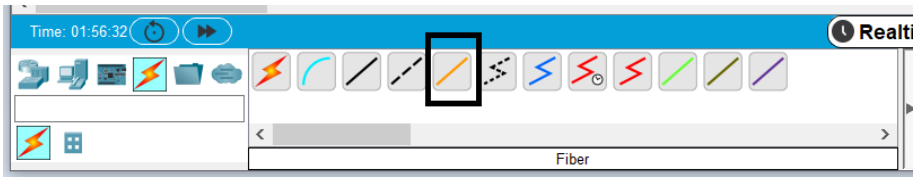
Фиг. 7.5. *Око-диаграма на филтрирания модулиран сигнал за два символни периода при rolloff фактор на RRC филтъра а) 0.3, б) 0.4 и в) 0.5*

7.3.2. Моделиране и конфигурация на оптични мрежови архитектури с помощта на симулатор Cisco Packet Tracer

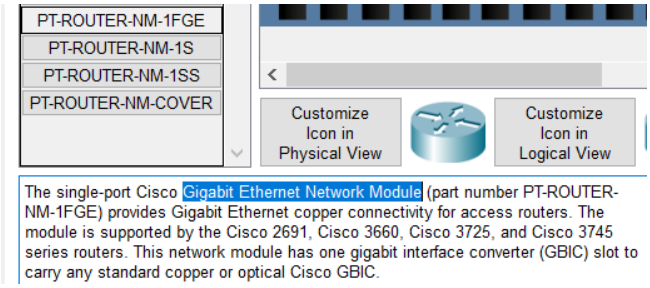
С въвеждането на оптични линии като преносни системи за информация в компютрите настъпва коренна промяна. Те имат някои съществени предимства, а именно:

- Предаването на данни е възможно да се извършва на големи разстояния без да е необходимо усилване и ретранслация на сигналите (отнася се за едномодовите оптични влакна);
- Става възможно пренасянето на голяма по обем информация от всяко оптично влакно;
- информационните канали на такива системи са предпазени от електрически смущения, тъй като материалите от които са направени са диелектрици. Това дава възможност да се използват за пренасяне на информация до обекти при които са налице силни електромагнитни смущения;
- Тъй като е невъзможна външна нежелателна интервенция, те са по-сигурни относно подслушване и нежелателна информационна интервенция;
- Полагането на оптичните кабели като информационни канали и материалите, от които са направени, са много по-евтини от медните.

Една от възможностите за симулационно моделиране, конфигурация и тестване на пакетното предаване на данни при оптични кабелни структури е средата Cisco Packet Tracer. За да може да бъде използвана fiber cable връзка, показана на фиг. 7.6, по отношение на маршрутизатори, комутатори, персонални компютърни станции или други мрежови устройства, трябва да бъде включен специализиран мрежови модул *Gigabit Ethernet Network Module* - (фиг. 7.7). Модулът позволява свързването на устройствата да се извърши посредством „*меден*“ или „*оптичен*“ тип интерфейс.



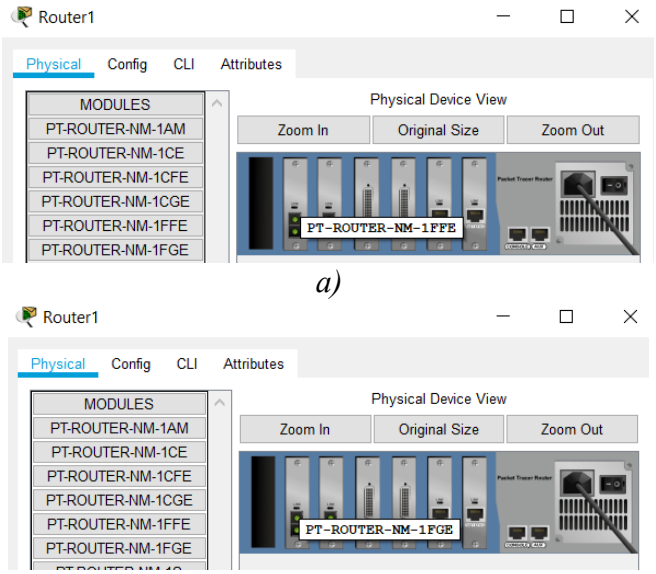
Фиг. 7.6. *Fiber интерфейс*



Фиг. 7.7. *Gigabit Ethernet Network Module*

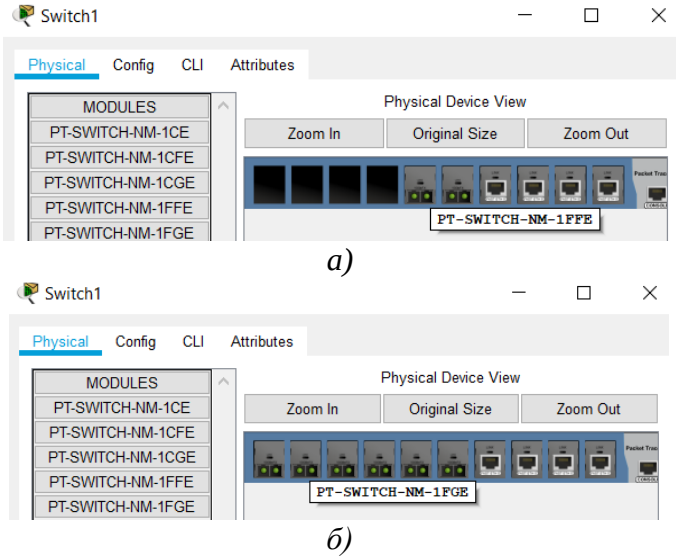
За целта първоначално ще бъде изградена мрежова архитектура с включване на маршрутизатор тип Router-PT, комутаторно звено тип Switch-PT и няколко потребителски станции Laptop 1, PC1, Laptop 2 и PC2. Преди свързване на всяко от устройствата първонално се интегрира указания по-горе специализиран модул, както следва при:

- Router-PT се премахват модулните Fast-Ethernet интерфейси (модул PT-ROUTER-NM-1FFE) и поставят PT-ROUTER-NM-1FGE модулни звена – фиг. 7.8;



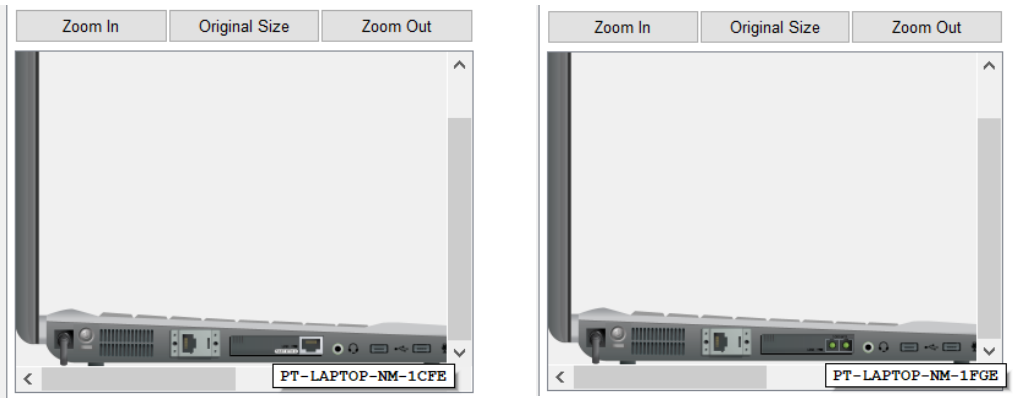
Фиг. 7.8. *Замяна на а) PT-ROUTER-NM-1FFE с б) PT-ROUTER-NM-1FGE модул*

- Switch-PT по подобен начин се извършва замяна на Fast-Ethernet PT-SWITCH-NM-1FFE и Gigabit Ethernet PT-SWITCH-NM-1FGE интерфейси, покано на фиг. 7.9;



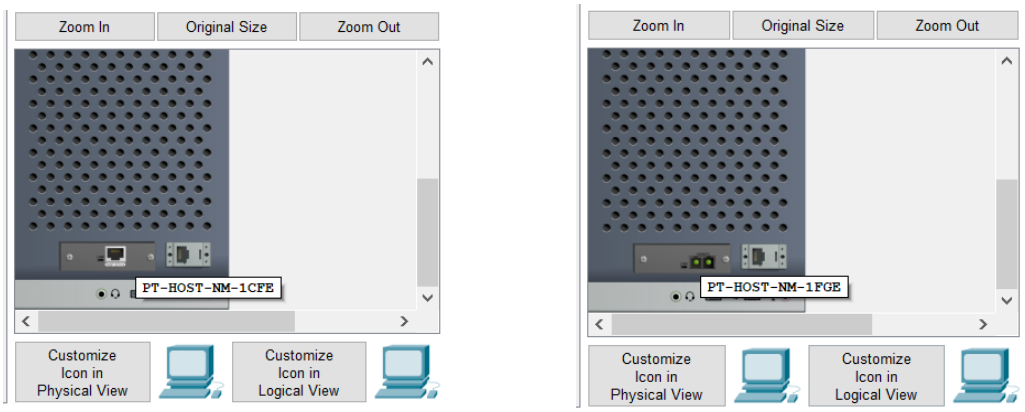
Фиг. 7.9. *Подмяна на модул а) PT-SWITCH-NM-1FFE с б) PT-SWITCH-NM-1FGE модул*

- Към Laptop и PC мрежови потребителски станции, следвайки аналогични процедури, се включват модули за връзка по меден/оптичен проводник PT-LAPTOP-NM-1FGE и PT-HOST-NM-1FGE, както е представено на фиг. 7.10 и фиг. 7.11;



а) *б)*

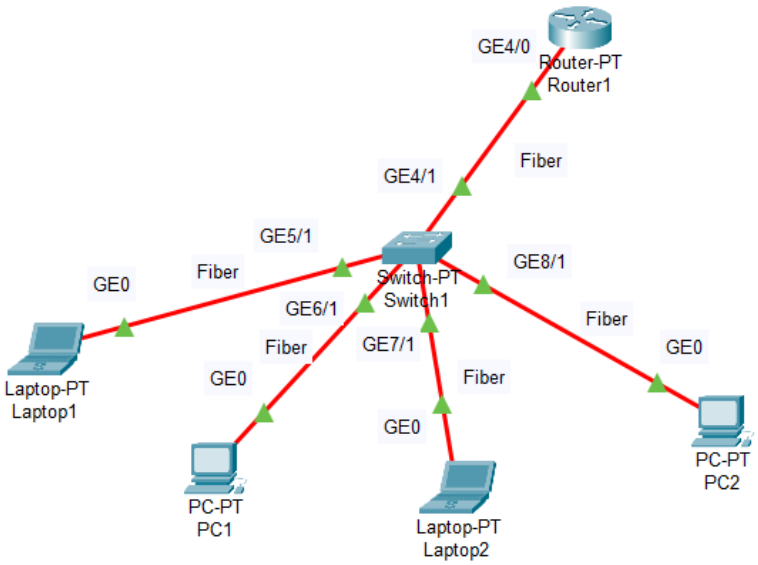
Фиг. 7.10. *Замяна на а) PT-LAPTOP-NM-1CFE с б) PT-LAPTOP-NM-1FGE модул*



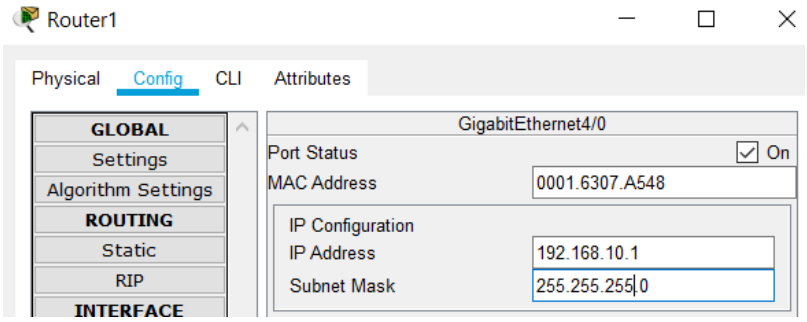
а) *б)*

Фиг. 7.11. *Замяна на а) PT-HOST-NM-1CFE с б) PT-HOST-NM-1FGE модул*

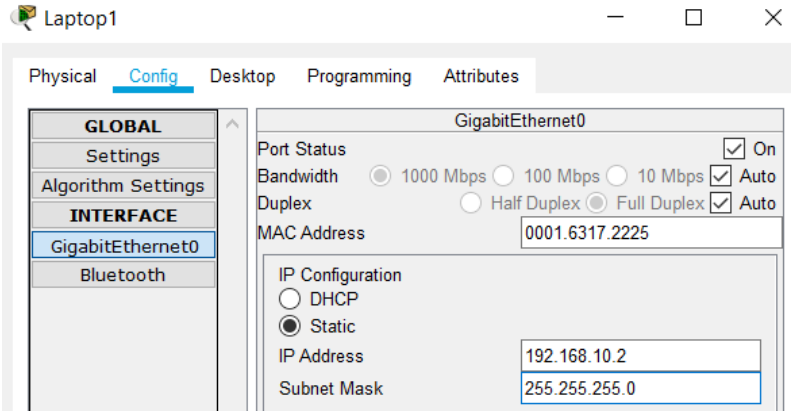
След включване на новите модули може да се премине към свързване на мрежовите устройства като се следва указаната топология, както и активиране (статус On) на обозначените връзки при всяко устройство на фиг. 7.12. На следващ етап трябва да се направи конфигурация на мрежата или установяване на логическа връзка между устройствата, свързана с назначаване на IP адреси и подмрежови маски по начина, показан на фиг. 7.13.



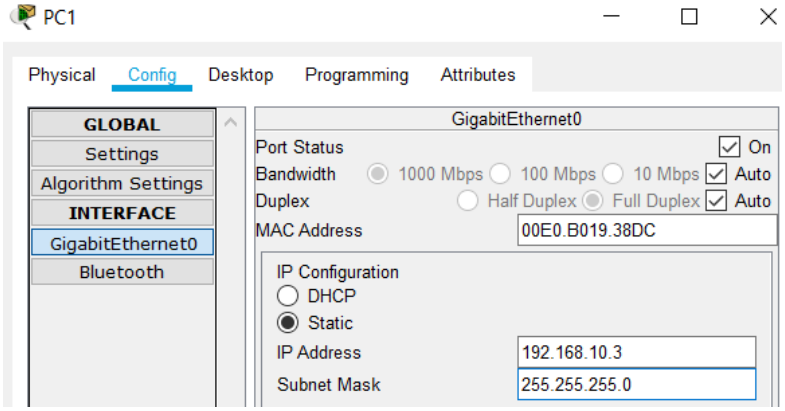
Фиг. 7.12. *Свързване на мрежови устройства посредством Fiber Optic Cables*


Router1
Physical
Config
CLI
Attributes
GLOBAL
Settings
Algorithm Settings
ROUTING
Static
RIP
INTERFACE
GigabitEthernet4/0
Port Status
On
MAC Address
0001.6307.A548
IP Configuration
IP Address
192.168.10.1
Subnet Mask
255.255.255.0

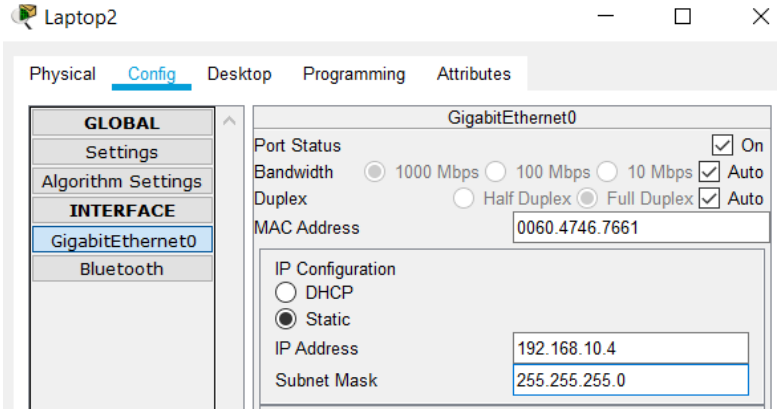
а)


Laptop1
Physical
Config
Desktop
Programming
Attributes
GLOBAL
Settings
Algorithm Settings
INTERFACE
GigabitEthernet0
Bluetooth
GigabitEthernet0
Port Status
On
Bandwidth
1000 Mbps
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
MAC Address
0001.6317.2225
IP Configuration
DHCP
Static
IP Address
192.168.10.2
Subnet Mask
255.255.255.0

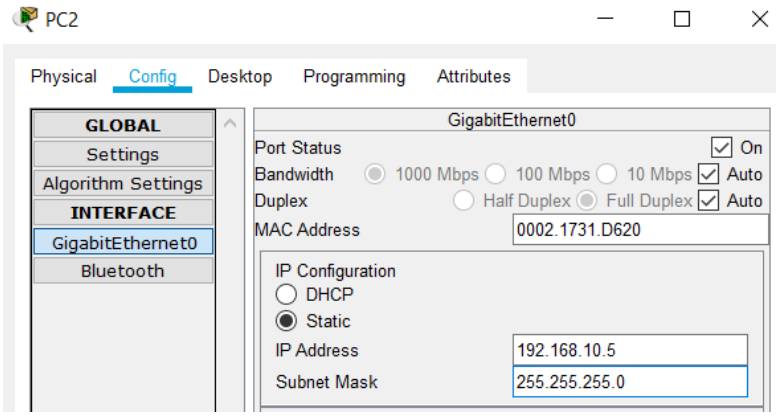
б)


PC1
Physical
Config
Desktop
Programming
Attributes
GLOBAL
Settings
Algorithm Settings
INTERFACE
GigabitEthernet0
Bluetooth
GigabitEthernet0
Port Status
On
Bandwidth
1000 Mbps
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
MAC Address
00E0.B019.38DC
IP Configuration
DHCP
Static
IP Address
192.168.10.3
Subnet Mask
255.255.255.0

в)

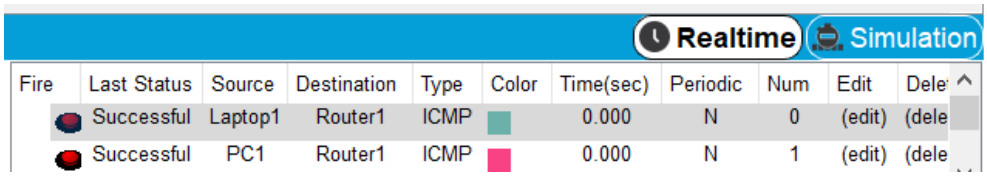

Laptop2
Physical
Config
Desktop
Programming
Attributes
GLOBAL
Settings
Algorithm Settings
INTERFACE
GigabitEthernet0
Bluetooth
GigabitEthernet0
Port Status
On
Bandwidth
1000 Mbps
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
MAC Address
0060.4746.7661
IP Configuration
DHCP
Static
IP Address
192.168.10.4
Subnet Mask
255.255.255.0

г)

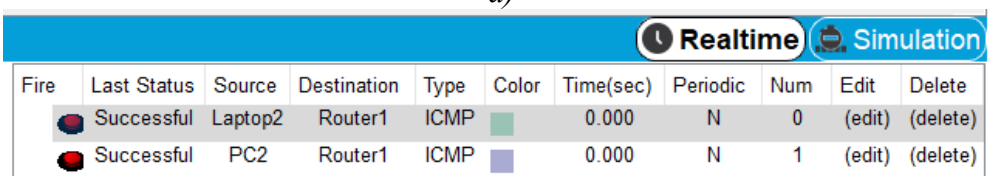


д)

Фиг. 7.13. *Конфигуриране на устройствата в оптичната мрежа*



а)



б)

Фиг. 7.14. *Тестване на пакетното предаване на данни между а) Laptop 1/PC1 с Router 1 и б) Laptop 2/PC2 с Router 1 устройствата в оптичната мрежа*

Последната стъпка по отношение на изградената оптична мрежова архитектура се състои в проверка и проследяване на надеждността на предаването на данни между мрежовите устройства. Процедурата може да бъде проведена в Realtime режим чрез изпращане на PDU информационни единици по указани от потребителя маршрути и преглед на статуса на Toggle PDU Window, както е представено на фиг. 7.14.

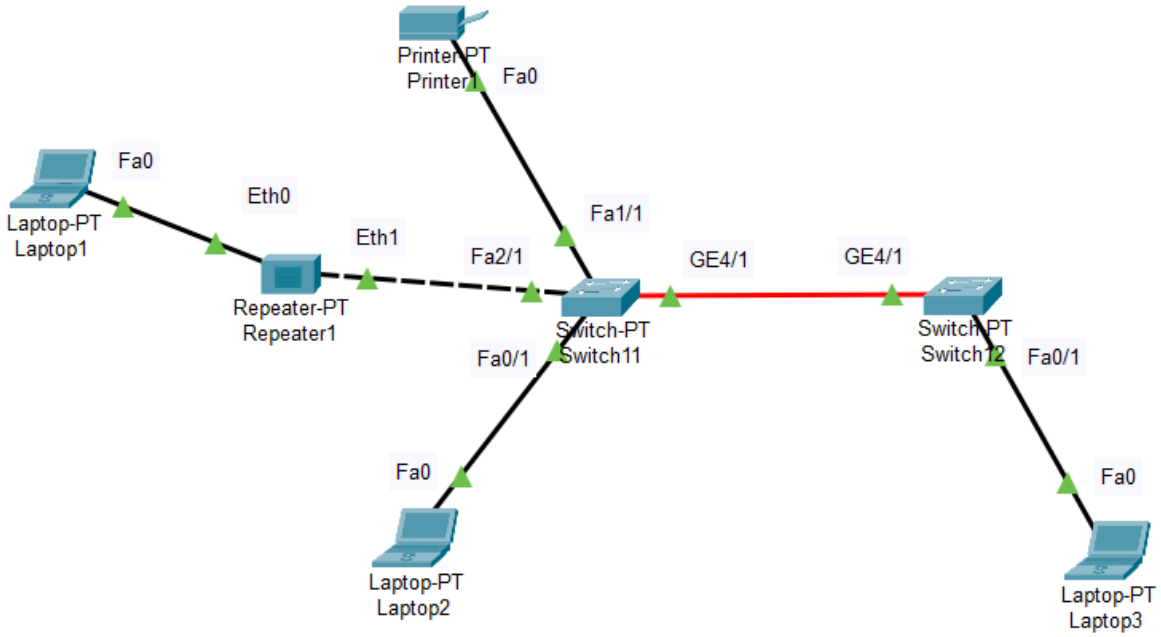
На следващ етап е предвидено да бъде разгледана ***оптична мрежова конфигурация*** с:

- *Кобнирано свързване с медни проводници и оптичен кабел;*
- *Добавяне на „Мрежови принтер“ и модул „Повторител“;*
- *Информационен трансфер, базиран на стандарт IPv6.*

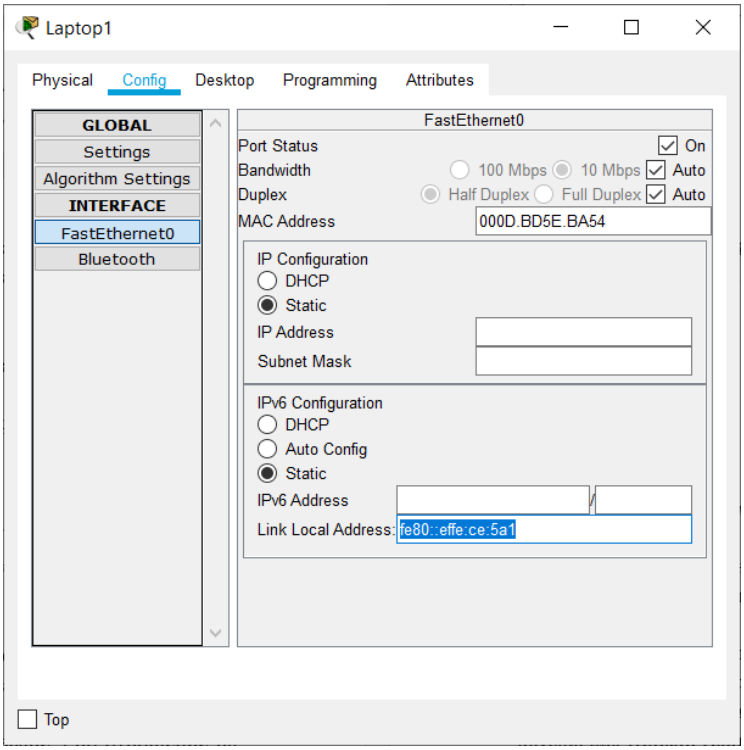
Така дефинираната архитектура, включваща два комутатора, мрежови принтер, повторител и три персонални потребителски станции, е показана на фиг. 7.15. Направени са следните видове свързвания:

- Copper Straight-Through (меден проводник „право свързване“) по отношение на “Laptop2 и Switch11”; “Laptop3 и Switch12”; “Switch11 и Printer1” чрез Fast Ethernet интерфейси;
- Copper Straight-Through по Fast Ethernet от Laptop1 към Ethernet интерфейс на Repeater-PT;
- Copper Cross-Over (меден проводник „обратно свързване“) чрез Fast Ethernet интерейси между Repeater-PT и Switch11;

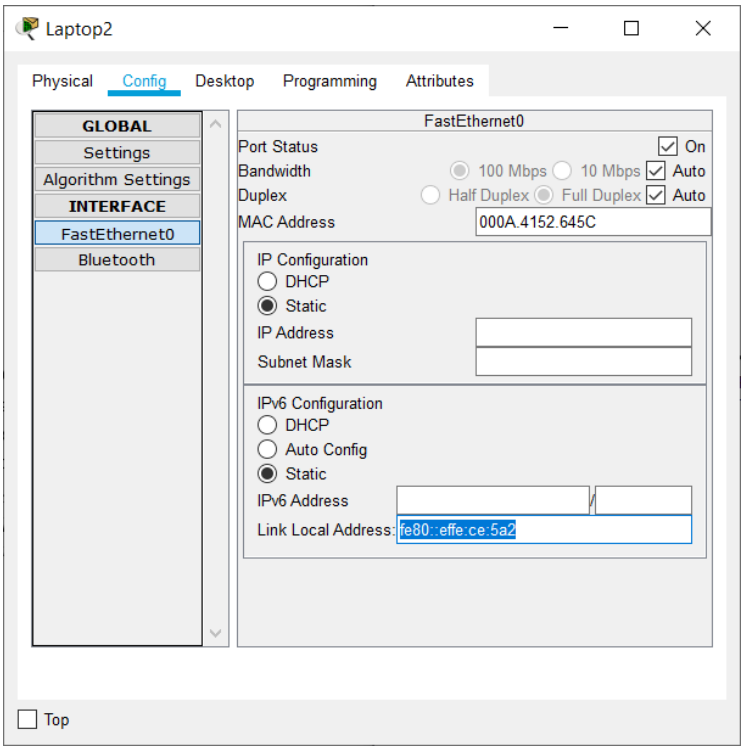
- *Fiber* между комутатори Switch11 и Switch12, използвайки Gigabit Ethernet интерфейси, за които е извършено включване на модули PT-SWITCH-NM-1FGE.



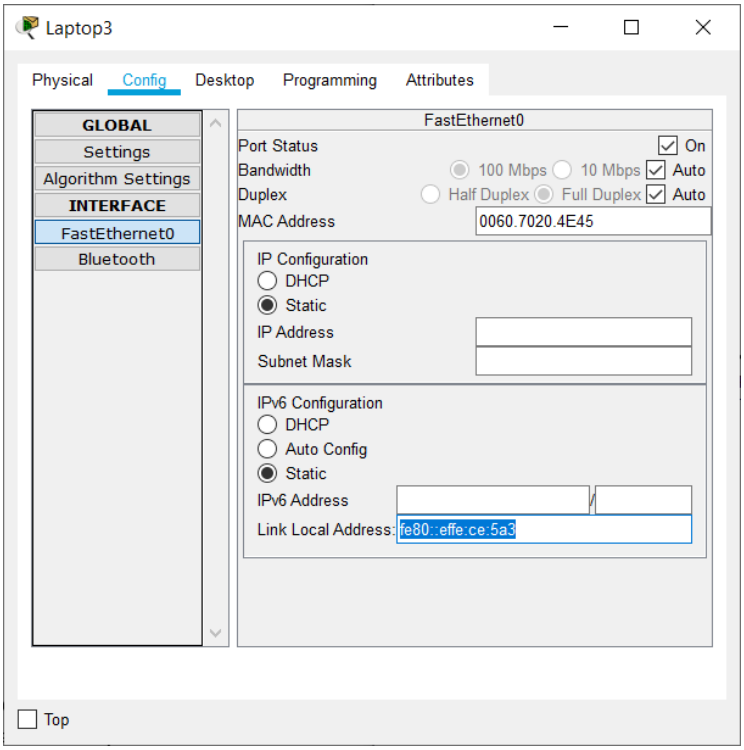
Фиг. 7.15. *Мрежова архитектура с включване на оптичен кабел, повторител и пакетно предаване на данни по IPv6 протокол*



а)

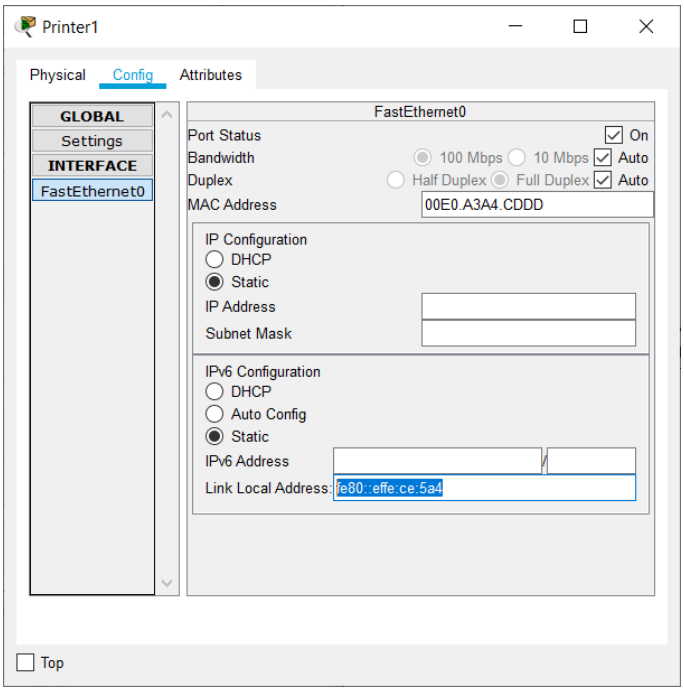

Laptop2
Physical
Config
Desktop
Programming
Attributes
GLOBAL
Settings
Algorithm Settings
INTERFACE
FastEthernet0
Bluetooth
FastEthernet0
Port Status
On
Bandwidth
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
MAC Address
000A.4152.645C
IP Configuration
DHCP
Static
IP Address
Subnet Mask
IPv6 Configuration
DHCP
Auto Config
Static
IPv6 Address
Link Local Address
fe80::effe:ce:5a2
Top

б)


Laptop3
Physical
Config
Desktop
Programming
Attributes
GLOBAL
Settings
Algorithm Settings
INTERFACE
FastEthernet0
Bluetooth
FastEthernet0
Port Status
On
Bandwidth
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
MAC Address
0060.7020.4E45
IP Configuration
DHCP
Static
IP Address
Subnet Mask
IPv6 Configuration
DHCP
Auto Config
Static
IPv6 Address
Link Local Address
fe80::effe:ce:5a3
Top

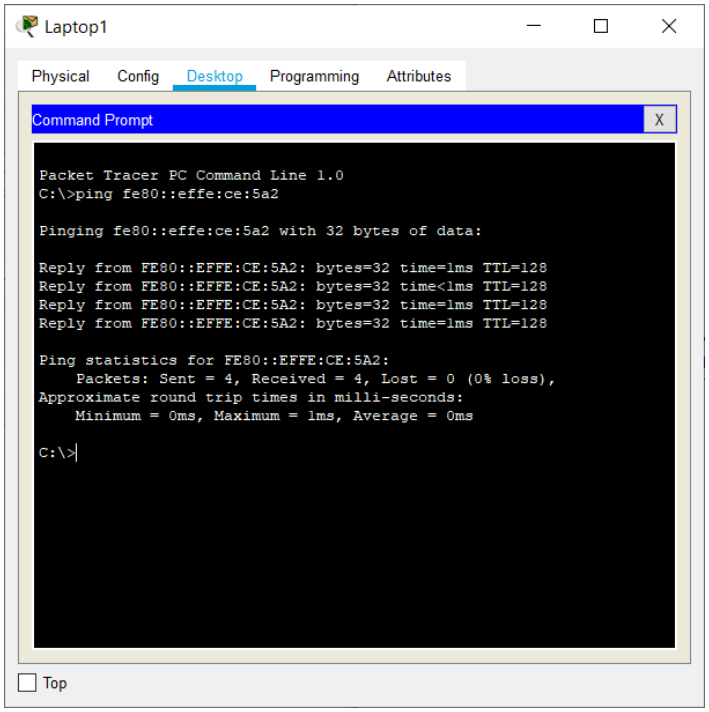
в)

След моделиране на мрежовата конфигурация с включване на оптичен кабел и мрежови повторител е необходимо да бъде направена конфигурация на компютърните потребителски станции и мрежовия принтер. Спезиализираната процедура се състои в назначаване на унифициран *Link Local Address* във връзка с *IP протокола версия 6. Спецификацията на тази версия на протокола определя записването на адреса да се извършва посредством низове от шестнадесетични числа, които са разделени с двуеточие.* Конфигурационните действия са представени на фиг. 7.16.

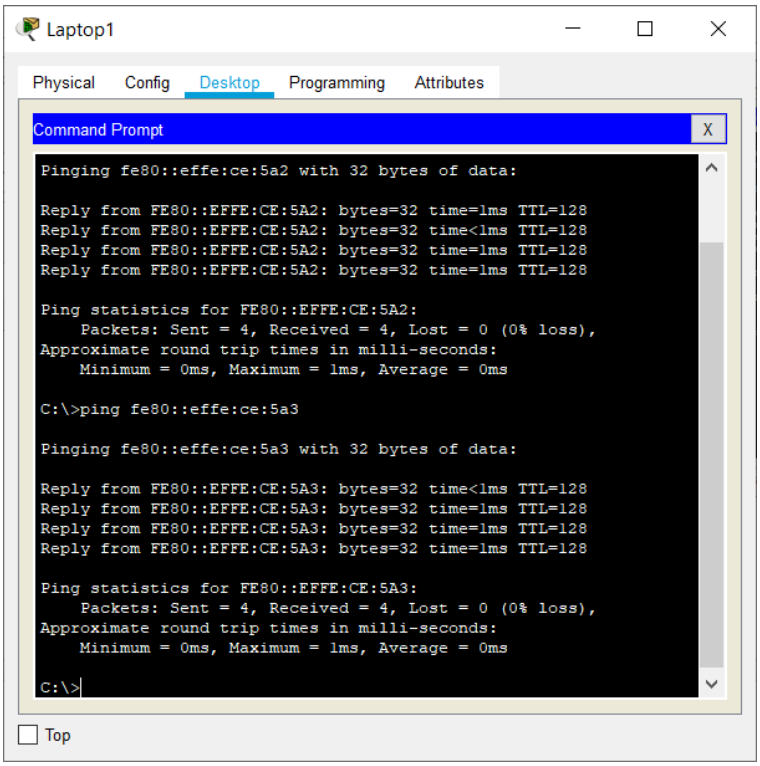


г)

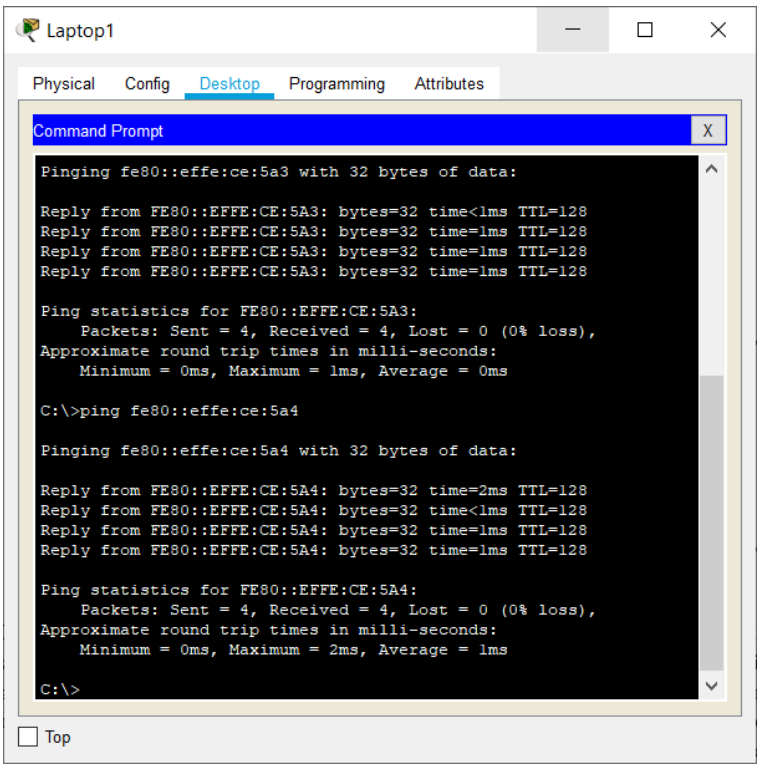
Фиг. 7.16. *Назначаване на адреси на персоналните компютърни станции и мрежовия принтер относно IPv6 протокол*



а)



б)

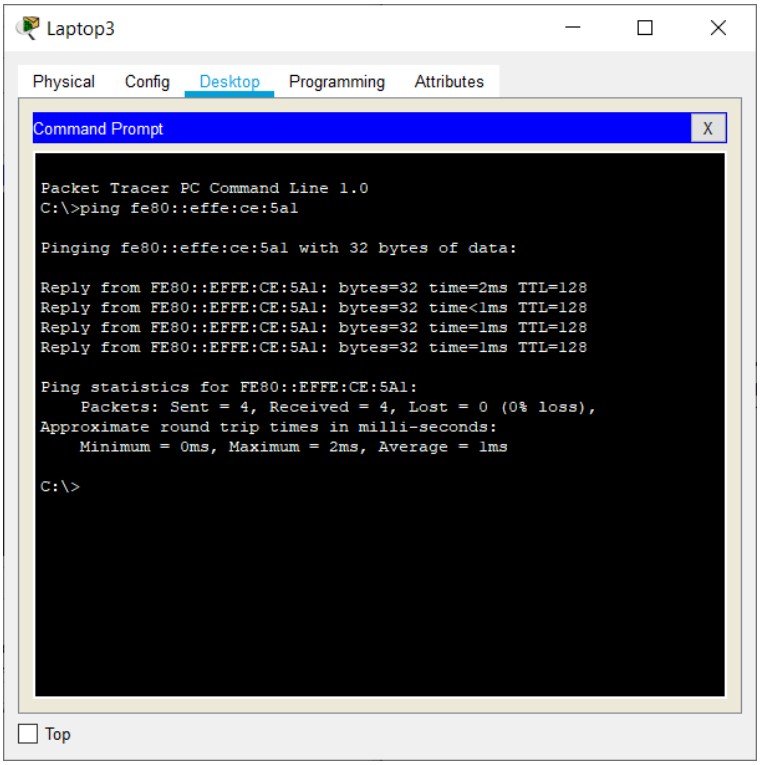


в)

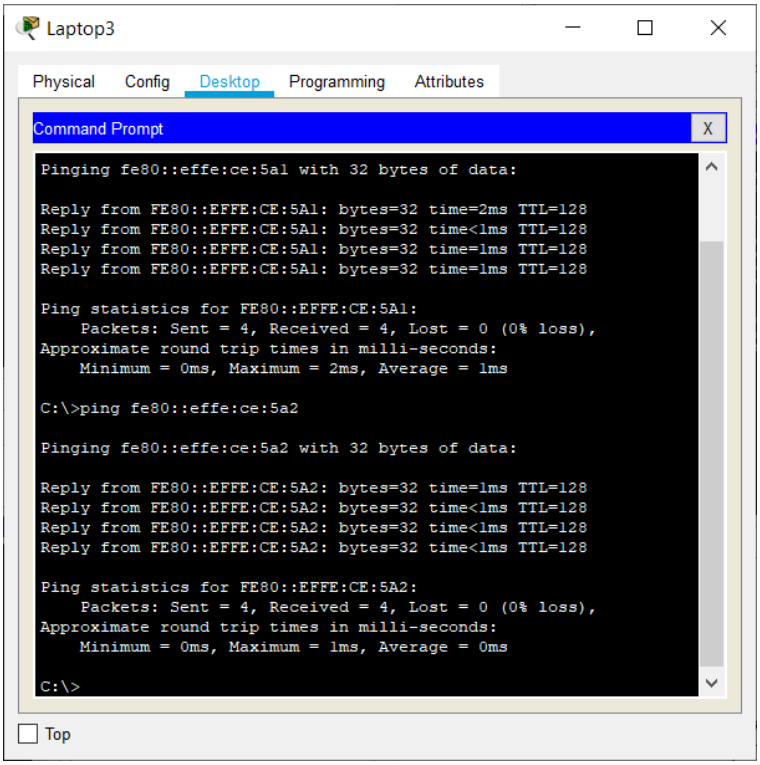
Фиг. 7.17. *Тестване на предаването на данни между Laptop1, персоналните станции „2 и 3“ и мрежовия принтер - а) “Laptop1 и Laptop2”, б) “Laptop1 и Laptop3” и в) “Laptop1 и Printer1”*

Последната стъпа от изграждането на мрежовата архитектура се отнася до проверка на коректността при предаването на пакетни данни между крайните мрежови възли. На

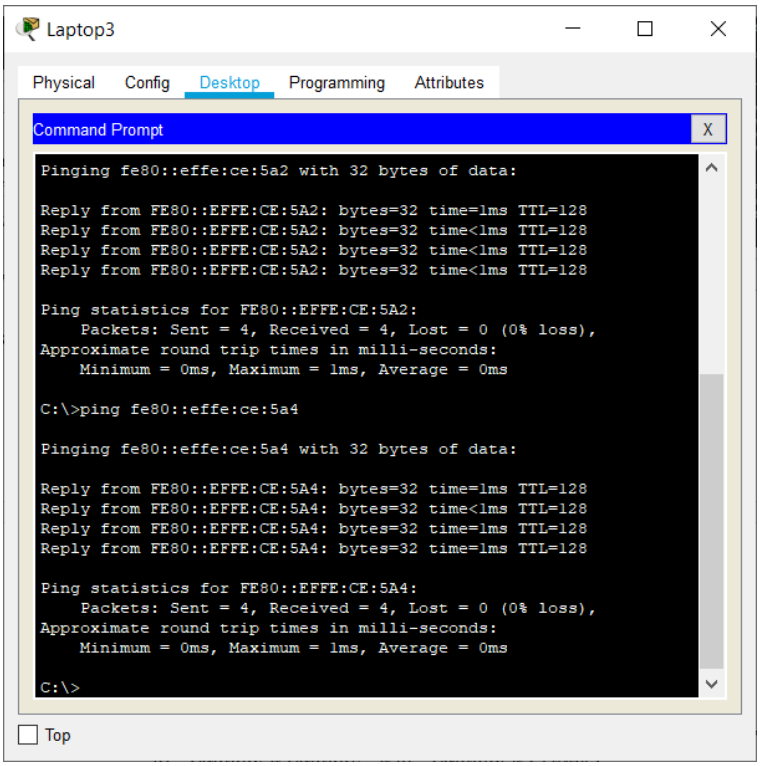
фиг. 7.17 са илюстрирани резултатите от тестването на комуникацията и информационния трансфер през *Command Prompt приложението* от Laptop1 към останалите две обособени потребителски станции и мрежовия принтер. Докато на фиг. 7.18 са проведени аналогични процеси относно станция Laptop3. Както може да се проследи, процесите по пакетно предаване са напълно коректни.



а)



б)



в)

Фиг. 7.18. *Тестване на предаването на данни между Laptop3, персоналните станции „1 и 2“ и мрежовия принтер - а) “Laptop3 и Laptop1”, б) “Laptop3 и Laptop2” и в) “Laptop3 и Printer1”*

7.4. Контролни въпроси.

1. Какви са заложените цели в Националния план за инфраструктура за достъп от следващо поколение (NGN)?
2. Какви са предимствата на оптичните технологии?
3. С какво название е известен RRC филтъра и с какво се свързва неговото приложение в цифровите модулации?
4. Какво представляват параметрите SNR и AWGN?
5. В какъв специфичен формат се задават IP адресите при използване на стандарта IP Version 6?

Лабораторно упражнение № 8

ДИМАМИЧНО МАРШРУТИЗИРАНЕ НА ОСНОВАТА НА RIP И OSPF ПРОТОКОЛИ В CISCO PACKET TRACER

8.1. Цел на упражнението

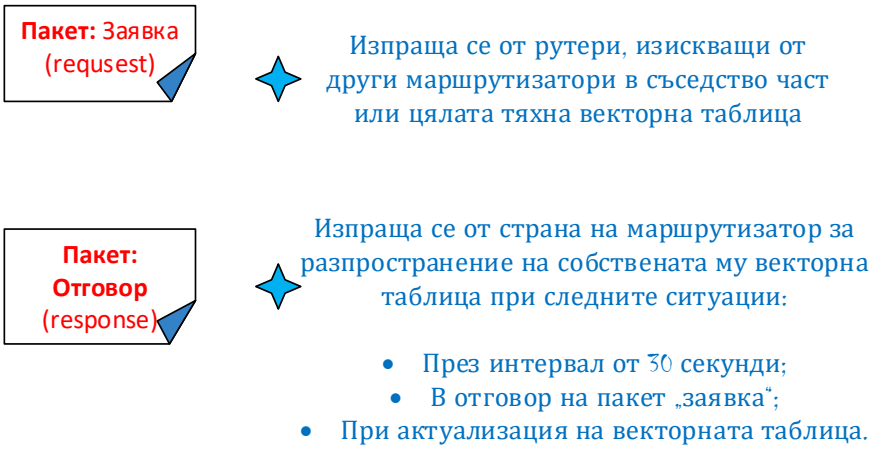
Да се запознаят студентите със спецификата и особеностите при прилагане на динамична маршрутизация за предаване на пакети с използване на специализираните комуникационни протоколи RIP и OSPF.

8.2. Теоретична постановка

Протокол за динамично маршрутизиране RIP

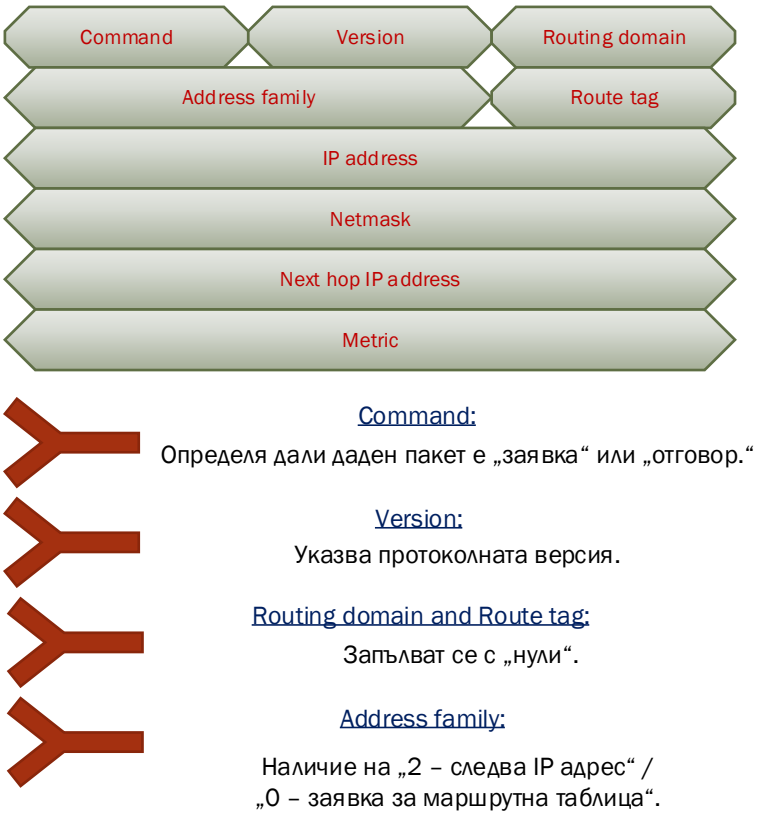
RIP или Routing Information Protocol е векторен маршрутизиращ протокол, който може да бъде причислен към групата на вътрешните протоколи IGP (Interior Gateway Protocol). Функционалността му се базира на Xerox PUP и XNS протоколите. Предназначен е предимно за малки компютърни мрежи. Основната метрика за разстояние, която се използва при стандарта, е броя „хопове“, достигащи до 15 в структурата на една RIP мрежа. Характеристика на протокола е изпращането на копие на маршрутната таблица през интервал от 30 секунди по направление към съседните маршрутизатори. Друга особеност се свързва с определяне на времевия интервал, след който даден маршрут до конкретен рутер се регламентира като „невалиден“, след което следва оказване на съседните маршритизатори да преустановят връзка с него. Промяната на мрежовата топология налага на рутерите преизчисляване на своите „вектори“. С цел повишаване на скоростта на сходимост се използва подходът „split horizon“, поради което се редуцира вероятността за възникване на цикли в маршрутите. Съществуват две основни версии на разглеждания стандарт, съответно [15]:

- Версия ***RIPv1***: Дефинирана е от RFC 1058. Има отношение към прилагане на “classful” тип маршрутизация, т.е. всички подмрежови сегменти, принадлежащи към даден клас, е необходимо да бъдат с еднакви маски. Реализира “broadcast” процедури по разпространение на маршрутна специфицирана информация, т.е. налице е обновяване на маршрутната таблица на всеки 30 секунди. Фигура 8.1 дава по-пълна представа за поддържаните типове пакети, респективно „заявка“ и „отговор“;



Фиг. 8.1. *Типове пакети в RIP стандарта*

- Версия ***RIPv2***: Характеризира се с поддържане на VLSM (Variable Length Subnet Masks – адресиране с променлива дължина на маските) и CIDR (Classless Interdomain Routing – безкласово IP адресиране) процедури. Тук е въведена мрежова автентикация с използване на „явен текст“, подобрена чрез криптографски хеш подход MD5 Message-Digest Algorithm или „дефиниция на дайджест на съобщенията 5“. На фиг. 8.2 е показан форматът на пакетите при RIPv2;



Фиг. 8.2. *Формат на пакетите при RIPv2*

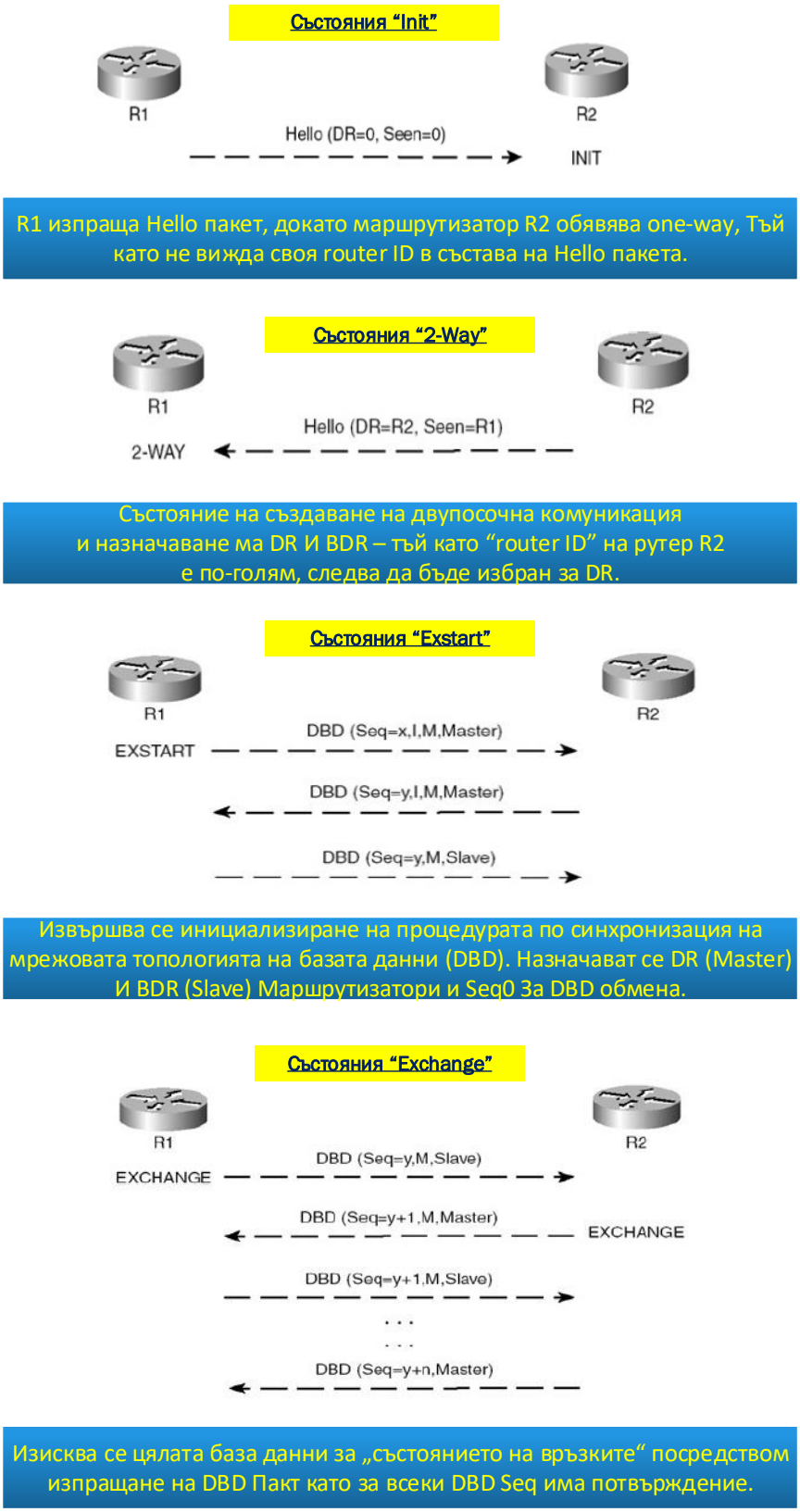
- Версия ***RIPng***: Представлява разширено и функционално надграждане на RIPv2 стандарта. Създаден е за поддържане на IPv6. Особеност на стандарта е, че не поддържа удостоверяване на актуализациите на версия RIPv1. Също така се изискват процедури по специфична кодировка на следващия „хоп“ за набор от записи на маршрути. Използва „multicast ff02::9” маршрутизация и изпраща обновления на UDP порт 521.

OSPF протокол за динамична мрежова маршрутизация

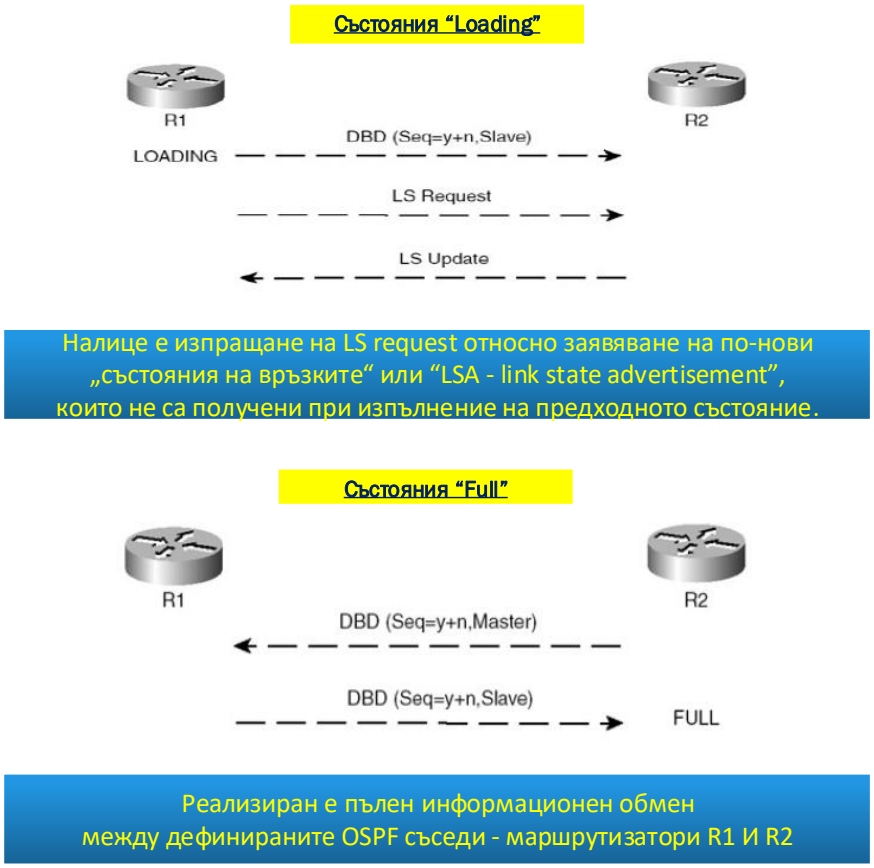
OSPF или Open Shortest Path First е IGP стандарт за динамична маршрутизация със следене на състоянието на връзката като изгражда цялостна топология на мрежата от обособени “areas”. Широко разпространен е за големи корпоративни мрежи и големи доставчици на мрежови услуги. Поддържа VLSM и CIDR като съставя маршрутни таблици по “destination IP” адресите в IP пакетите. Осъществява бърза конвергация при възникване на структурни топологични промени [16].

Топологията на протокола наподобява „граф“, при която с помощта на Dijkstra's algorithm се извършва изчисляване на „дървото на най-късия път или shortest path tree“. OSPF функционира посредством създаване на „съседство“ между маршрутизатори, между които има директна връзка и периодично изпращане на „Hello“ пакетни данни.

Структурните рутери реализират избор на Designated Router (DR) и Backup Designated Router (BDR), използван като концентратор. Налице е изпращане на пакети и „link state updates“ чрез “unicast” и “multicast” тип комуникация между устройствата. Фигура 8.3 онагледява основните състояния на OSPF стандарта за динамична маршрутизация.



а)



б)

Фиг. 8.3. *OSPF състояния – а) "Init", "2-Way", "Exstart" и "Exchange" и б) "Loading" и "Full"*

Особеност на OSPF е директната употреба на IP protocol 89 като за разлика от BGP стандарта не се базира на TCP или UDP. Използва "Multicast" адреси 224.0.0.5 за „SPF/link state рутери, AllSPFRouters" и 224.0.0.6, т.е. „Designated рутери и AllDRouters". Разполага със собствени механизми за откриване и коригиране на грешките.

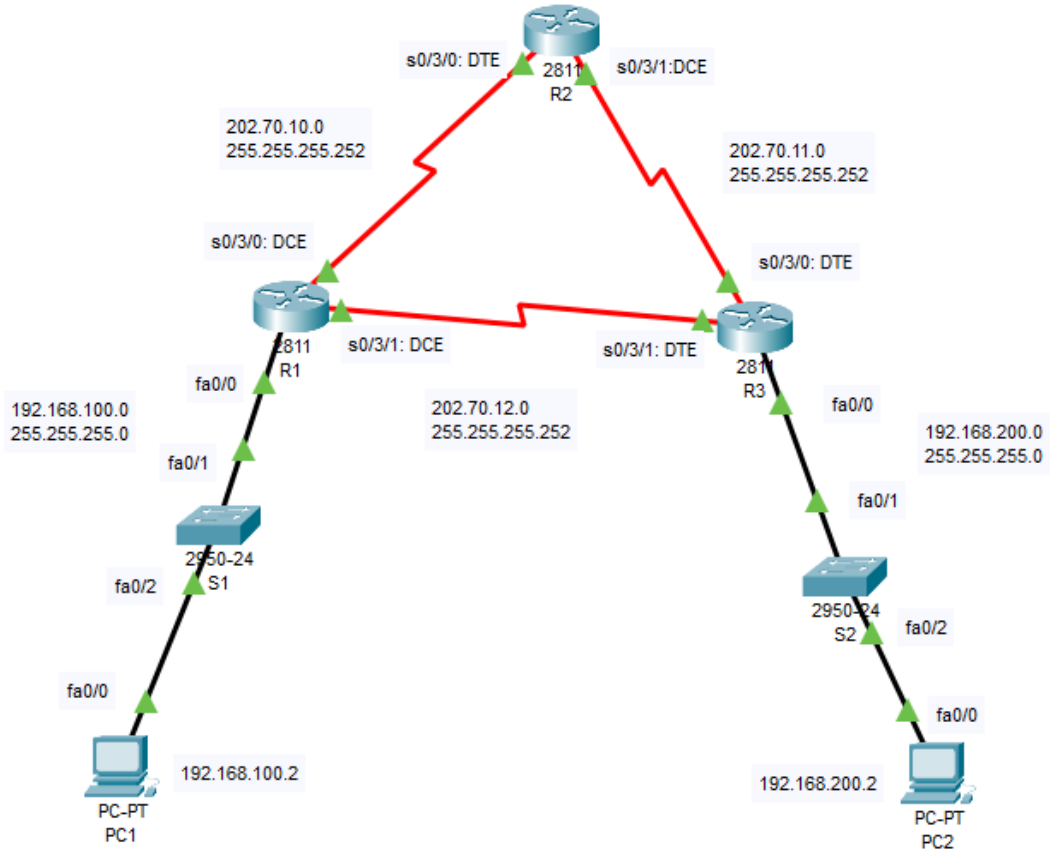
Изградената OFPF мрежа се разделя на области, обозначавани чрез 32-битови специфицирани индикатори, подобни на IPv4 адреси. При IPv6 се прилагат същите означения. По отношение на записа на идентификаторите се препоръзва използването на десетично-точковата нотация. Самите области се разглеждат като логически групи от хостове и мрежи. Всяка област поддържа собствена "link state database", която се представя в обобщен вид. Всички области се свързват към „опорната" или "backbone area – 0.0.0.0" с помощта на физическа връзка и виртуален канал. Процесите по маршрутизация се базират на „гранични рутери" или Area Border Routers (ABRs), свързани в опорната и в някоя от другите области.

Посредством Autonomous System Border Router (ASBR) се реализира връзката между "area 0.0.0.0" на OSPF мрежата с „външния свят". По дефиниция дадена OSPF мрежа с всички принадлежащи ѝ обособени области е известна под названието „Автономна система" [15, 16].

8.3. Последователност на работа и задачи за изпълнение

8.3.1. Изграждане и управление на мрежова конфигурация с динамично предаване на пакети с помощта на RIP и PPP протоколи

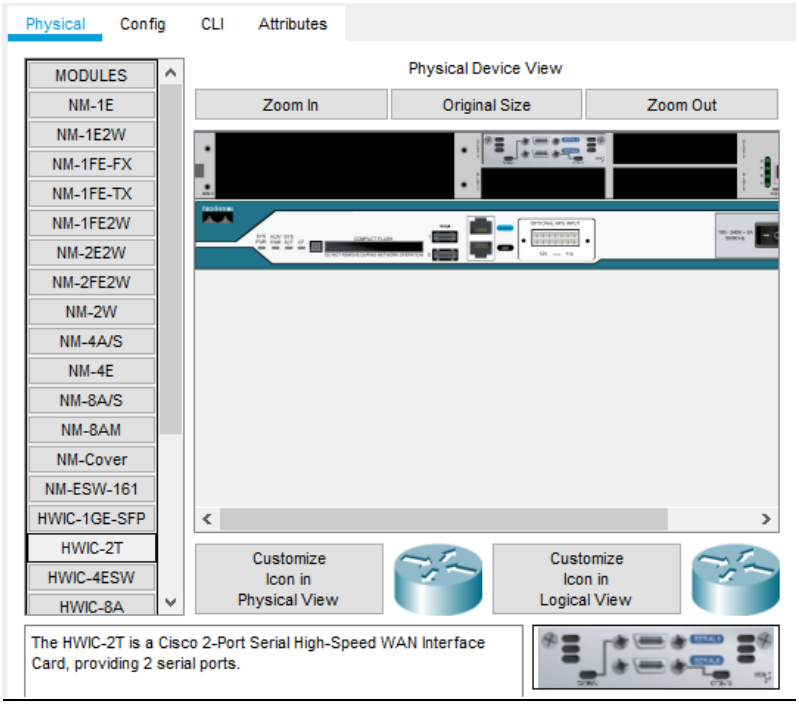
По-долу, на фиг. 8.4 е представена възможна мрежова конфигурация с конфигуриране на RIP-based динамично предаване на данни между три мрежови маршрутизатора, свързани чрез серийна комуникация.



Фиг. 8.4. *Мрежова конфигурация за динамично предаване на пакети с приложение на RIP и PPP протоколи*

В схемната конфигурация са включени още кумутатори и стационарни потребителски компютърни станции. Особеност при свързването между мрежовите устройства по серийната комуникационна връзка е спазването на страна DCE или DTE, т.е. „комуникационно“ или „терминално“ вид оборудване, по отношение на възможността за фиксиране на скоростта на трансфер на пакети. Изграждането и конфигурацията на мрежовата комуникация между участващите устройства преминава през следната стъпкова последователност:

- Добавяне на комуникационен модул HWIC 2T за включване на серийни комуникационни интерфейси при мрежови маршрутизатори, обозначени с R1, R2 и R3 (фиг. 8.5);



Фиг. 8.5. *Включване на модул за установяване на серийна комуникация HWIC-2T*

- Конфигуриране на Fast Ethernet и Serial типове комуникационни интерфейси при мрежови маршрутизатори R1, R2 и R3. Процесите включват:

  - ✓ задаване на IP адрес при подмрежова маска при Fa0/0;
  - ✓ процедури по капсулиране на данните чрез *PPP протокол* и задаване на скорост на трансфер 64000 bps (параметричната скорост може да бъде фиксирана при интерфейс относно страна “DCE”);
  - ✓ назначаване на мрежовите направления на динамична маршрутизация посредством *RIP протокол*.

Указаните дейности се извършват чрез въвеждане на конфигурационни команди посредством CLI (Command Line Interface). Процесите са последователно показани от фиг. 8.6 до фиг. 8.13;

- Interface fa0/0 при R1:

```
Router>enable
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fa0/0
Router(config-if)#ip address 192.168.100.1 255.255.255.0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up

Router(config-if)#exit
```

Фиг. 8.6. *IP конфигурация на интерфейс fa0/0 при R1*

- Interfaces s0/3/0 и s0/3/1 при R1:

```
Router(config)#interface s0/3/0
Router(config-if)#ip address 202.70.10.1 255.255.255.252
Router(config-if)#encapsulation ppp
Router(config-if)#clock rate 64000
Router(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/3/0, changed state to down
Router(config-if)#exit
```

а)

```
Router(config)#interface s0/3/1
Router(config-if)#ip address 202.70.12.1 255.255.255.252
Router(config-if)#encapsulation ppp
Router(config-if)#clock rate 64000
Router(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/3/1, changed state to down
Router(config-if)#exit
```

б)

Фиг. 8.7. *Настройка на серийни интерфейси при маршрутизатор R1 - а) s0/3/0 и б) s0/3/1*

- RIP routing при R1:

```
Router(config)#router rip
Router(config-router)#network 192.168.100.0
Router(config-router)#network 202.70.10.0
Router(config-router)#network 202.70.12.0
Router(config-router)#exit
```

Фиг. 8.8. *Назначаване на мрежи за RIP динамична маршрутизация при рутер R1*

- Interfaces s0/3/0 и s0/3/1 при R2:

```
Router>enable
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface s0/3/0
Router(config-if)#ip address 202.70.10.2 255.255.255.252
Router(config-if)#encapsulation ppp
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface Serial0/3/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/3/0, changed
state to up

Router(config-if)#exit
```

а)

```
Router(config)#interface s0/3/1
Router(config-if)#ip address 202.70.11.1 255.255.255.252
Router(config-if)#encapsulation ppp
Router(config-if)#clock rate 64000
Router(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/3/1, changed state to down
Router(config-if)#exit
```

б)

Фиг. 8.9. *Конфигурация на серийни интерфейси при маршрутизатор R2 - а) s0/3/0 и б) s0/3/1*

- RIP routing при R2:

```
Router(config)#router rip
Router(config-router)#network 202.70.10.0
Router(config-router)#network 202.70.11.0
Router(config-router)#exit
```

Фиг. 8.10. *Назначаване на мрежи за RIP динамична маршрутизация при рутер R2*

- Interface fa0/0 при R3:

```
Router>enable
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fa0/0
Router(config-if)#ip address 192.168.200.1 255.255.255.0
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up

Router(config-if)#exit
```

Фиг. 8.11. *IP конфигурация на интерфейс fa0/0 при R3*

- Interfaces s0/3/0 и s0/3/1 при R3:

```
Router(config)#interface s0/3/0
Router(config-if)#ip address 202.70.11.2 255.255.255.252
Router(config-if)#encapsulation ppp
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface Serial0/3/0, changed state to up

Router(config-if)#exit
```

а)

```
Router(config)#interface s0/3/1
Router(config-if)#ip address 202.70.12.2 255.255.255.252
Router(config-if)#encapsulation ppp
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface Serial0/3/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/3/1, changed
state to up

Router(config-if)#exit
```

б)

Фиг. 8.12. *Настройка на серийни интерфейси при маршрутизатор R3 - а) s0/3/0 и б) s0/3/1*

- RIP routing при R3:

```
Router(config)#router rip
Router(config-router)#network 192.168.200.0
Router(config-router)#network 202.70.11.0
Router(config-router)#network 202.70.12.0
Router(config-router)#exit
```

Фиг. 8.13. *Дефиниране на мрежи за RIP динамична маршрутизация при рутер R3*

- Провеждане на проверка относно коректността на назначените рутери в структурата на обектните Routing Information Base или RIP Database, както и маршрутната таблица Routing Table. Стъпката се повтаря при маршрутизатори R1, R2 и R3, онагледено от фиг. 8.14 до фиг. 8.19:

- RIP Database при R1:

```
Router>show ip rip database
192.168.100.0/24    auto-summary
192.168.100.0/24    directly connected, FastEthernet0/0
192.168.200.0/24    auto-summary
192.168.200.0/24
    [1] via 202.70.12.2, 00:00:01, Serial0/3/1
202.70.10.0/30    auto-summary
202.70.10.0/30    directly connected, Serial0/3/0
202.70.11.0/24    auto-summary
202.70.11.0/24
    [1] via 202.70.10.2, 00:00:26, Serial0/3/0    [1] via
202.70.12.2, 00:00:01, Serial0/3/1
202.70.12.0/30    auto-summary
202.70.12.0/30    directly connected, Serial0/3/1
```

Фиг. 8.14. *Предглед на RIP Database при маршрутизатор R1*

- Routing Table при R1:

```
Router>show ip route rip
R    192.168.200.0/24 [120/1] via 202.70.12.2, 00:00:20, Serial0/3/1
     202.70.10.0/24 is variably subnetted, 2 subnets, 2 masks
R    202.70.11.0/24 [120/1] via 202.70.10.2, 00:00:15, Serial0/3/0
                    [120/1] via 202.70.12.2, 00:00:20, Serial0/3/1
```

Фиг. 8.15. *Проверка на маршрутната таблица при рутер R1*

- RIP Database при R2:

```
Router>show ip rip database
192.168.100.0/24    auto-summary
192.168.100.0/24
    [1] via 202.70.10.1, 00:00:15, Serial0/3/0
192.168.200.0/24    auto-summary
192.168.200.0/24
    [1] via 202.70.11.2, 00:00:26, Serial0/3/1
202.70.10.0/30    auto-summary
202.70.10.0/30    directly connected, Serial0/3/0
202.70.11.0/30    auto-summary
202.70.11.0/30    directly connected, Serial0/3/1
202.70.12.0/24    auto-summary
202.70.12.0/24
    [1] via 202.70.10.1, 00:00:15, Serial0/3/0    [1] via
202.70.11.2, 00:00:26, Serial0/3/1
```

Фиг. 8.16. *Предглед на RIP Database при маршрутизатор R2*

- Routing Table при R2:

```
Router>show ip route rip
R    192.168.100.0/24 [120/1] via 202.70.10.1, 00:00:04, Serial0/3/0
R    192.168.200.0/24 [120/1] via 202.70.11.2, 00:00:14, Serial0/3/1
     202.70.11.0/24 is variably subnetted, 2 subnets, 2 masks
R    202.70.12.0/24 [120/1] via 202.70.10.1, 00:00:04, Serial0/3/0
                    [120/1] via 202.70.11.2, 00:00:14, Serial0/3/1
```

Фиг. 8.17. *Проверка на маршрутната таблица при рутер R2*

- RIP Database при R3:

```
Router>show ip rip database
192.168.100.0/24    auto-summary
192.168.100.0/24
    [1] via 202.70.12.1, 00:00:24, Serial0/3/1
192.168.200.0/24    auto-summary
192.168.200.0/24    directly connected, FastEthernet0/0
202.70.10.0/24    auto-summary
202.70.10.0/24
    [1] via 202.70.11.1, 00:00:02, Serial0/3/0    [1] via
202.70.12.1, 00:00:24, Serial0/3/1
202.70.11.0/30    auto-summary
202.70.11.0/30    directly connected, Serial0/3/0
202.70.12.0/30    auto-summary
202.70.12.0/30    directly connected, Serial0/3/1
```

Фиг. 8.18. *Предглед на RIP Database при маршрутизатор R3*

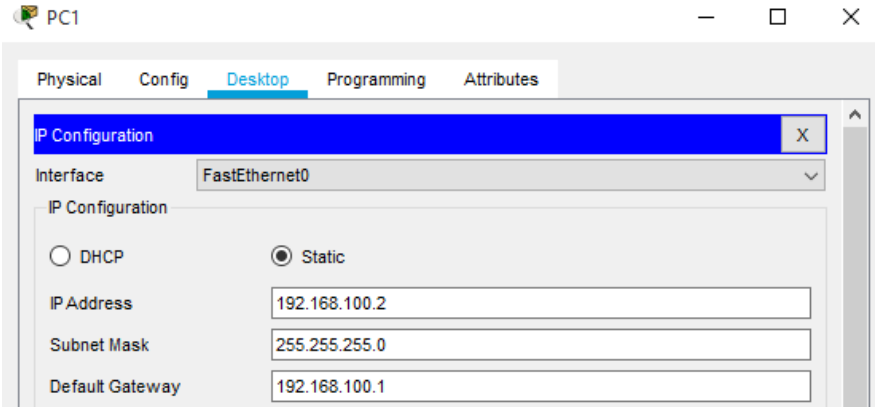
- Routing Table при R3:

```
Router>show ip route rip
R    192.168.100.0/24 [120/1] via 202.70.12.1, 00:00:19, Serial0/3/1
R    202.70.10.0/24 [120/1] via 202.70.11.1, 00:00:25, Serial0/3/0
                    [120/1] via 202.70.12.1, 00:00:19, Serial0/3/1
```

Фиг. 8.19. *Проверка на маршрутната таблица при рутер R3*

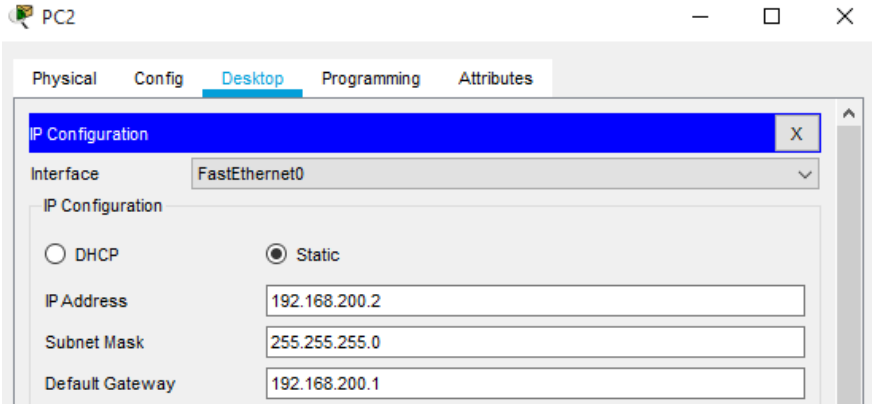
- Създаване на логическа свързаност между персоналните потребителски компютърни станции PC1 и PC2. Продурата, дадена на фиг. 8.20 и фиг. 8.21, касае задаване на IP адрес, подмрежова маска и мрежови шлюз или Gateway:

- IP адрес и Subnet mask при PC1:



Фиг. 8.20. *IP конфигурация на компютърна станция PC1*

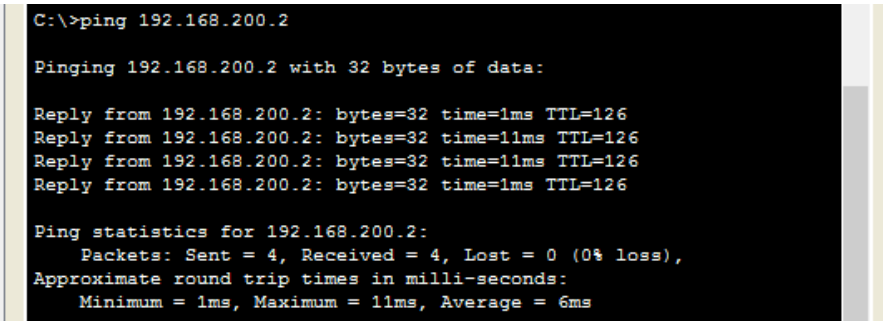
- IP адрес и Subnet mask при PC2:



Фиг. 8.21. *IP конфигурация на компютърна станция PC2*

Тестване на коректността при предаване на данни между PC1 и PC2 на базата на създадените мрежови условия за реализация на RIP динамична маршрутизация. Проверката се свежда до назначаване на направления за предаване на PDU единици с помощта на ping процедура:

- ping от PC1 към PC2:

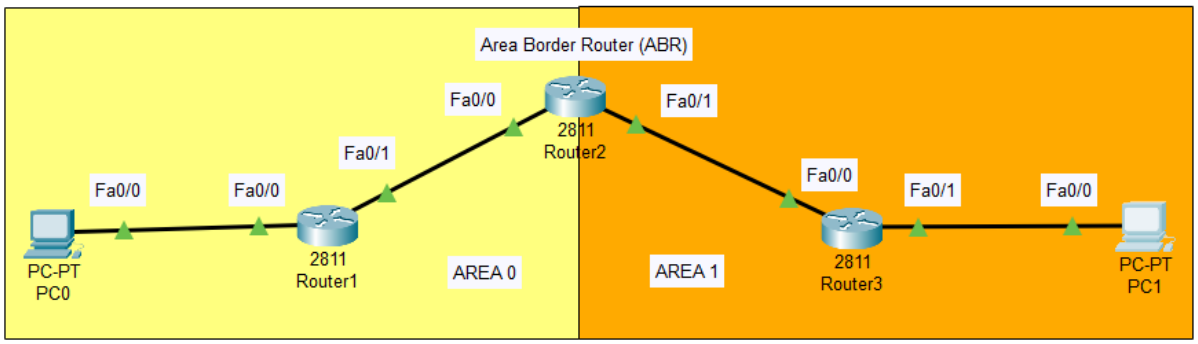


Фиг. 8.22. *Ping процедура от станция PC1 към мрежови потребител PC2*

В конкретния случай на фиг. 8.22 е демонстрирано успешното предаване на четири пакета от данни по направление от PC1 към PC2. Процедурата може да бъде повторена по обратното направление при идентична ефективност.

8.3.2. Моделиране на мрежова конфигурация с интеграция на динамично предаване на пакети на базата на OSPF

В случая може да бъде разгледана OSPF мрежовата конфигурация за предаване на пакетни данни между две обособени логически групи чрез използване на единичен специализиран Area Border Router, представена на фиг. 8.23.



Фиг. 8.23. *Конфигурация с динамично предаване на пакети между две области на основата на OSPF протокол*

Настройката и конфигурацията на мрежовите устройства, протича в следната систематизирана последователност:

- ❖ IP и Interface конфигуриране на маршрутизаторите относно fa0/0 и fa0/1 от фиг. 8.24 до фиг. 8.26:

- Маршрутизатор R1:

```
Router>enable
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R1
R1(config)#interface fa0/0
R1(config-if)#ip address 192.168.0.1 255.255.255.0
R1(config-if)#no shutdown

R1(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up

R1(config-if)#interface fa0/1
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#no shutdown

R1(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up

R1(config-if)#exit
```

Фиг. 8.24. *Конфигурация на интерфейси fa0/0 и fa0/1 при R1*

- Маршрутизатор R2:

```
Router>enable
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R2
R2(config)#interface fa0/0
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config-if)#no shutdown

R2(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up

R2(config-if)#exit
R2(config)#interface fa0/1
R2(config-if)#ip address 192.168.2.1 255.255.255.0
R2(config-if)#no shutdown

R2(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up

R2(config-if)#exit
```

Фиг. 8.25. *Настройка на интерфейси fa0/0 и fa0/1 при рутер R2*

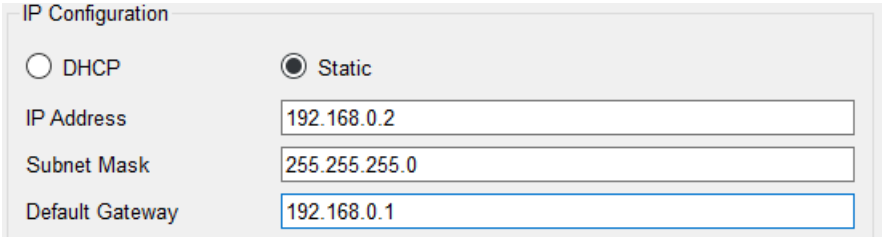
- Маршрутизатор R3:

```
Router>enable
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R3
R3(config)#interface fa0/0
R3(config-if)#ip address 192.168.2.2 255.255.255.0
R3(config-if)#no shutdown

R3(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0,
changed state to up

R3(config-if)#exit
R3(config)#interface fa0/1
R3(config-if)#ip address 192.168.3.1 255.255.255.0
R3(config-if)#no shutdown

R3(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1,
changed state to up

R3(config-if)#exit
```

Фиг. 8.26. *Конфигурация на интерфейси fa0/0 и fa0/1 при маршрутизатор R3*

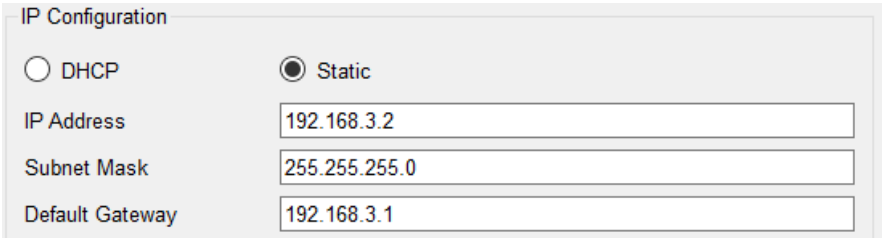
❖ Конфигуриране на персонални станции PC0 и PC1. Процедурат се отнася до назначаване на IP адрес, подмрежова маска и шлюз – фиг. 8.27 и фиг. 8.28:

- Потребителсна станция PC0:



Фиг. 8.27. *IP конфигурация на интерфейси на потребителска станция PC1*

- Персонална компютърна станция PC1:



Фиг. 8.28. *Настройка на IP Address, Subnet Mastк и Gateway при станция PC2*

❖ Назначаване на OSPF протокол на динамично маршрутизиране в обхват на Multy-area, т.е. включване на логически зони “Area 0” и “Area 1”. Дейностите се състоят в задаване на OSPF маршрутизация с идентификатор номер „1“ и фиксиране на специфичните протоколни мрежови параметри, респективно:

- “IP address”;
- “Wildcard mask” - позволява администриране на обновленията (Receive / Not Receive) относно маршрутизаторите спрямо отделните обособени мрежови сегменти. Маската е специфична за OSPF протокола и по дефиция е огледална на подмрежовата маска;
- “Area ID” – идентификатор на областта;
- “router-id” - служи за идентификация на устройството в OSPF мрежата.

- Дефиниране на OSPF протокол за R1, попадащ в обхвата на “Area 0” на фиг. 8.29:

```
R1>enable
R1#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router ospf 1
R1(config-router)#network 192.168.0.0 0.0.0.255 area 0
R1(config-router)#network 192.168.1.0 0.0.0.255 area 0
R1(config-router)#router-id 1.1.1.1
R1(config-router)#Reload or use "clear ip ospf process" command,
for this to take effect


R1(config-router)#exit
```

Фиг. 8.29. *Параметрична настройка на OSPF за маршрутизатор R1*

- Дефиниране на OSPF маршрутизация за R2, попадащ в обхвата на “Area 0” (Fa0/0) и “Area 1” (Fa0/1), показано на фиг. 8.30:

```
R2>enable
R2#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ospf 1
R2(config)#router ospf 1
R2(config-router)#network 192.168.1.0 0.0.0.255 area 0
R2(config-router)#network 192.168.2.0 0.0.0.255 area 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#Reload or use "clear ip ospf process" command,
for this to take effect


R2(config-router)#exit
```

Фиг. 8.30. *Конфигуриране на OSPF протокол при маршрутизатор R2*

- Задаване на OSPF маршрутизация за R3, попадащ в обхвата на “Area 1”, представено на фиг. 8.31:

```
R3>enable
R3#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#router ospf 1
R3(config-router)#network 192.168.2.0 0.0.0.255 area 1
R3(config-router)#network 192.168.3.0 0.0.0.255 area 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#Reload or use "clear ip ospf process" command,
for this to take effect


R3(config-router)#exit
```

Фиг. 8.31. *Конфигуриране на OSPF стандарт при маршрутизатор R3*

❖ OSPF верификация за маршрутизатор R2 относно установена “neighbor relationship” с маршрутизатори R1 и R3, дадена на фиг. 8.32:

```
R2>show ip ospf neighbor


Neighbor ID     Pri   State           Dead Time   Address
Interface
192.168.1.1       1   FULL/DR         00:00:38    192.168.1.1
FastEthernet0/0
192.168.3.1       1   FULL/BDR        00:00:38    192.168.2.2
FastEthernet0/1
```

Фиг. 8.32. *Верификция на R2 относно установяване на „neighbor relationship“ с R1 и R3*

❖ OSPF верификация за маршрутизатор R1 и R3 относно проверка за налична директна връзка между мрежите към всеки маршрутизатор в обхвата на различните области, показана на фиг. 8.33 и фиг. 8.34:

- Верификация за директна връзка по отношение на маршрутизатор R1:

```
R1>show ip route ospf
O IA 192.168.2.0 [110/2] via 192.168.1.2, 00:20:26,
FastEthernet0/1
O IA 192.168.3.0 [110/3] via 192.168.1.2, 00:16:11,
FastEthernet0/1
```

Фиг. 8.33. *Верификция на установена „директната връзка“ при рутер R1*

- Верификация за директна връзка по отношение на маршрутизатор R3:

```
R3>show ip route ospf
O IA 192.168.0.0 [110/3] via 192.168.2.1, 00:17:39,
FastEthernet0/0
O IA 192.168.1.0 [110/2] via 192.168.2.1, 00:17:39,
FastEthernet0/0
```

Фиг. 8.34. *Верификция на установена „директната връзка“ при рутер R3*

❖ Проверка на текущото състояние на маршрутната таблица – OSPF рутери (show ip route), направена от фиг. 8.35 до фиг. 8.37:

- Маршрутизатор R1:

```
R1>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter
area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external
type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E -
EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.0.0/24 is directly connected, FastEthernet0/0
C    192.168.1.0/24 is directly connected, FastEthernet0/1
O IA 192.168.2.0/24 [110/2] via 192.168.1.2, 00:23:42,
FastEthernet0/1
O IA 192.168.3.0/24 [110/3] via 192.168.1.2, 00:19:27,
FastEthernet0/1
```

Фиг. 8.35. *Верификция на маршрутната таблица при рутер R1*

- Маршрутизатор R2:

```
R2>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter
area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external
type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E -
EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

O    192.168.0.0/24 [110/2] via 192.168.1.1, 00:25:17,
FastEthernet0/0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/1
O    192.168.3.0/24 [110/2] via 192.168.2.2, 00:20:19,
FastEthernet0/1
```

Фиг. 8.36. *Проверка на маршрутната таблица за R2*

- Маршрутизатор R3:

```
R3>show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile,
B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter
area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external
type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E -
EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia -
IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

O IA 192.168.0.0/24 [110/3] via 192.168.2.1, 00:21:31,
FastEthernet0/0
O IA 192.168.1.0/24 [110/2] via 192.168.2.1, 00:21:31,
FastEthernet0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/0
C    192.168.3.0/24 is directly connected, FastEthernet0/1
```

Фиг. 8.37. *Верификация на маршрутната таблица при рутер R3*

❖ Системна верификация на статуса и обхвата на наличните текущите маршрутни протоколи (show ip protocols), проведена от фиг. 8.38 до фиг. 8.40:

- Маршрутизатор R1“:

```
R1>show ip protocols

Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.1.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    192.168.0.0 0.0.0.255 area 0
    192.168.1.0 0.0.0.255 area 0
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.1.1           110      00:28:36
    192.168.2.1           110      00:27:55
  Distance: (default is 110)
```

Фиг. 8.38. *Проверка на състоянието и обхвата на OSPF при рутер R1*

- Маршрутизатор R2:

```
R2>show ip protocols

Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.2.1
  Number of areas in this router is 2. 2 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    192.168.1.0 0.0.0.255 area 0
    192.168.2.0 0.0.0.255 area 1
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.1.1           110      00:29:46
    192.168.2.1           110      00:25:01
    192.168.3.1           110      00:24:46
  Distance: (default is 110)
```

Фиг. 8.39. *Верификация на състоянието и обхвата на OSPF при маршрутизатор R2*

- Маршрутизатор R3:

```
R3>show ip protocols

Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.3.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    192.168.2.0 0.0.0.255 area 1
    192.168.3.0 0.0.0.255 area 1
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.2.1           110      00:25:48
    192.168.3.1           110      00:25:33
  Distance: (default is 110)
```

Фиг. 8.40. *Верификция на състоянието и обхвата на OSPF при рутер R3*

- Ping процедура от PC0 и PC1:

```
C:\>ping 192.168.3.2

Pinging 192.168.3.2 with 32 bytes of data:

Reply from 192.168.3.2: bytes=32 time=1ms TTL=125
Reply from 192.168.3.2: bytes=32 time=2ms TTL=125
Reply from 192.168.3.2: bytes=32 time<1ms TTL=125
Reply from 192.168.3.2: bytes=32 time<1ms TTL=125

Ping statistics for 192.168.3.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 0ms
```

Фиг. 8.41. *Тестване на пакетното предаване посредством OSPF протокол между персонални компютърни станции PCO и PC1*

- Ping процедура по направление от PC1 към PC0:

```
C:\>ping 192.168.0.2

Pinging 192.168.0.2 with 32 bytes of data:

Reply from 192.168.0.2: bytes=32 time<1ms TTL=125
Reply from 192.168.0.2: bytes=32 time<1ms TTL=125
Reply from 192.168.0.2: bytes=32 time<1ms TTL=125
Reply from 192.168.0.2: bytes=32 time<1ms TTL=125

Ping statistics for 192.168.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Фиг. 8.42. *Тестване на пакетното предаване посредством OSPF протокол по направление между персонални компютърни станции PC1 и PC0*

❖ Тестване на коректността на конфигурационните настройки чрез изпълнение на назначени Ping процедури по основните информационни направления за пакетно предаване, реализирано на фиг. 8.41 и фиг. 8.42.

8.4. Контролни въпроси.

1. В какво се състоят различията между версиите RIPng, RIPv1 и RIPv2?
2. Коя е основната метрика при RIP комуникационния протокол?
3. Каква процедура се изпълнява през интервал от 30 сек. между мрежови маршрутизатори, базирани на комуникация по RIP протокол?
4. Какви са специфичните особености на стандарт за динамична маршрутизация OSPF?
5. Какво представляват назначените структурните „area 0 … N“ и как се свързат с „опорната“ или “backbone area“ при OSPF комуникационен протокол?

Лабораторно упражнение № 9

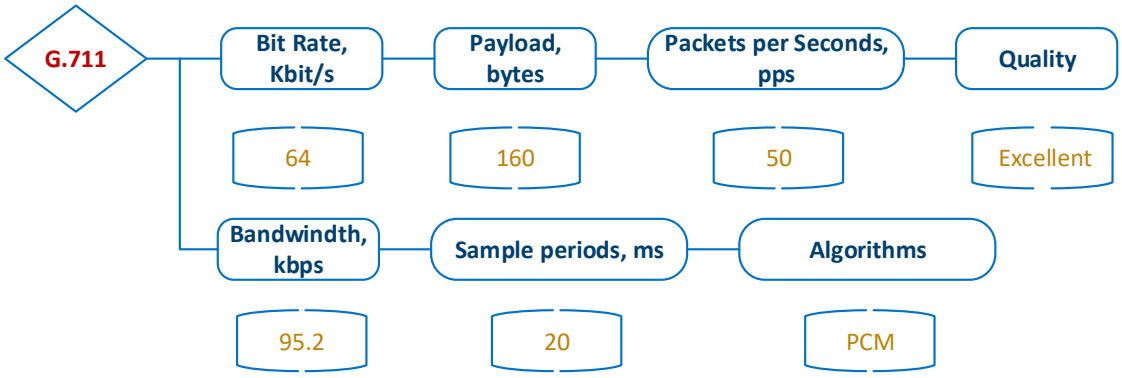
УПРАВЛЕНИЕ НА ТРАФИК ОТ ГЛАСОВИ ДАННИ НА БАЗА НА УСЛУГАТА VOIP В СРЕДА CISCO PACKET TRACER

9.1. Цел на упражнението

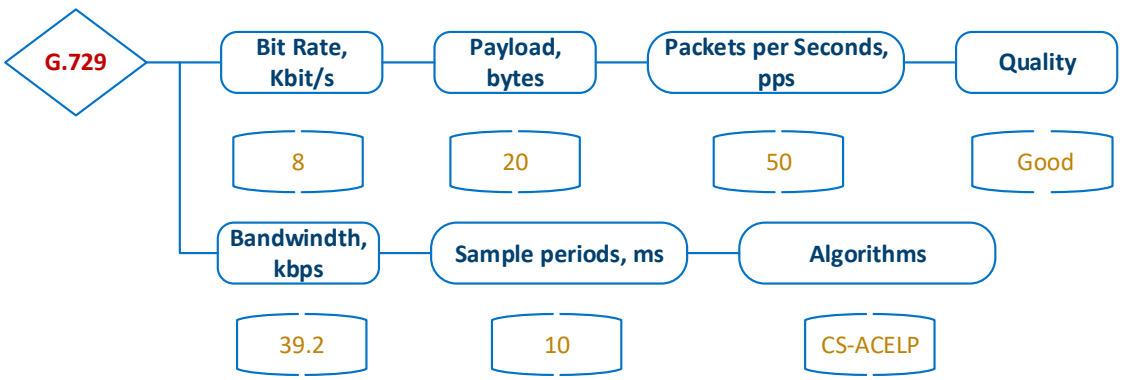
Да се запознаят студентите с технологията за пренос на гласови данни Voice over Internet Protocol посредством ресурсите на глобалната Интернет мрежа.

9.2. Теоретична постановка

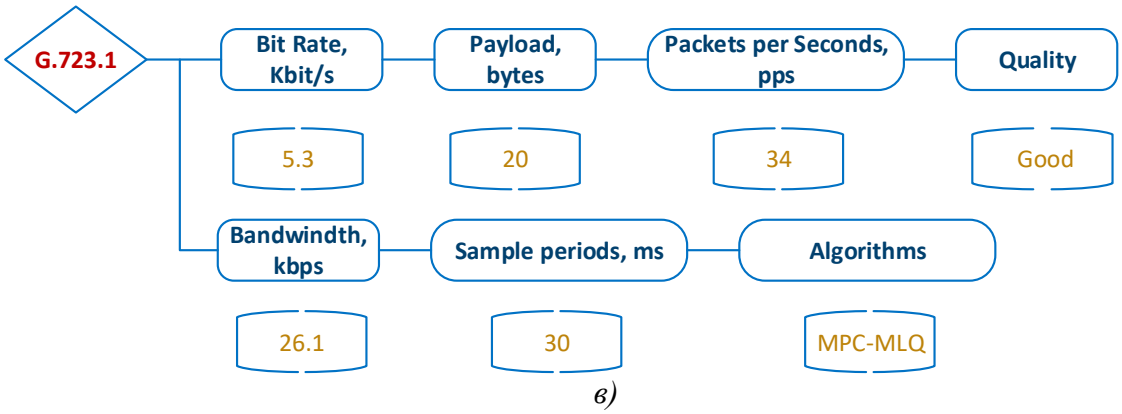
VoIP или ***Voice over Internet Protocol*** се разглежда като съвременна технология, служеща за конвертиране и компресиране на гласовия трафик в цифров формат, след което се енкапсулира под формата на IP пакети и транспортира посредством обществени или частни IP мрежи като Internet, Ethernet и Wireless LAN. Технологията позволява пренос на гласови пакети в реално време между кои да е две точки в струтурата на дадена IP мрежа, притежаващи валидни IP адреси. VoIP се базира на използването на различни типове компресиращи алгоритми. Тук се въвежда понятието „Codec“ или „compression - decompression“. Codecs представляват инструменти за конвертиране на аналогови сигнали като „видео“ или „глас“ към цифрови данни или преобразуване на цифровизиран поток към аналогова форма. По-долу на фиг. 9.1 е дадена сравнителна характеристика между някои от използваните Codecs и свързаните с тях компресиращи алгоритми в технологията [17].



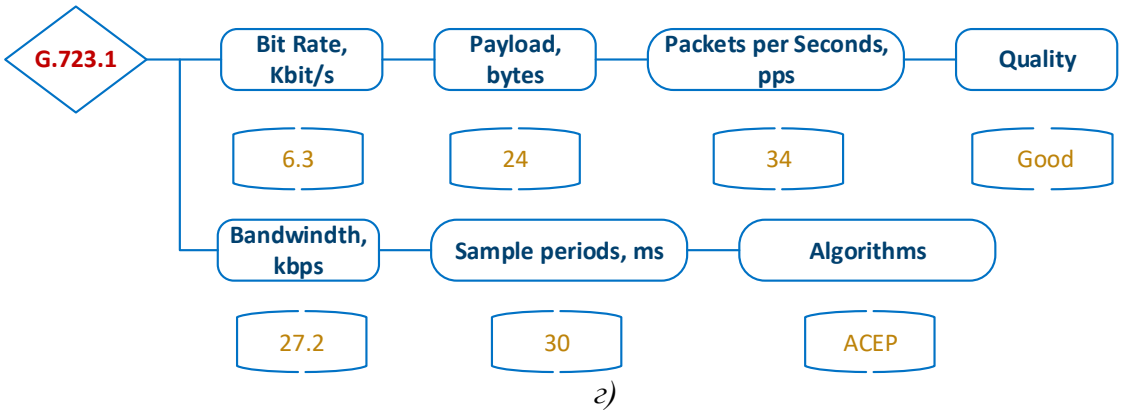
а)



б)

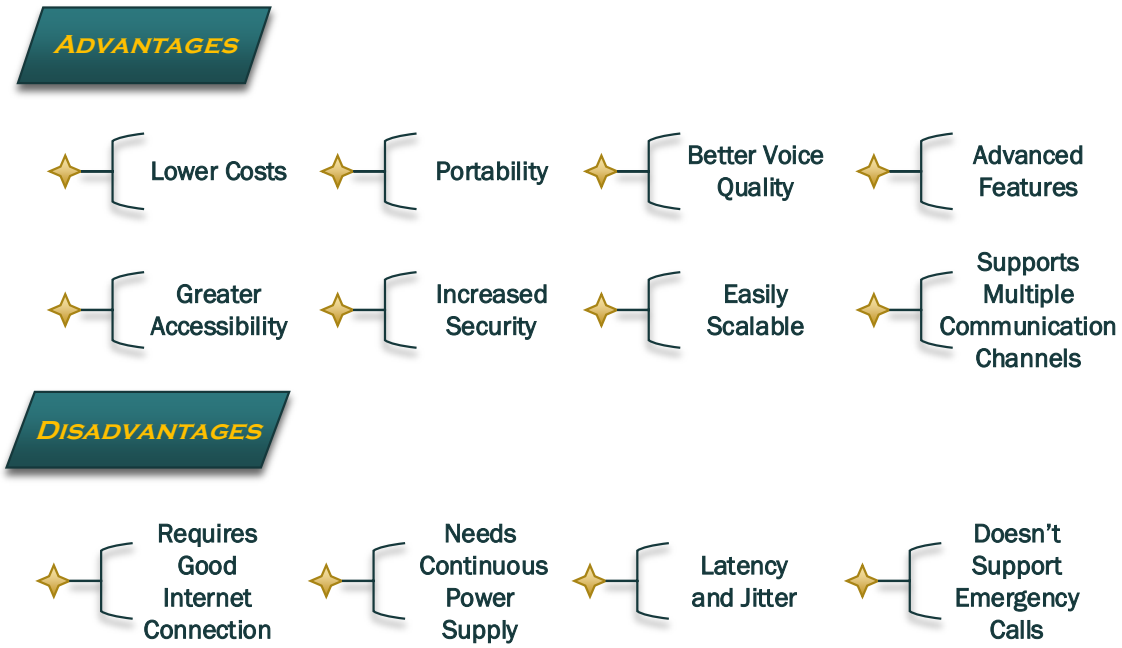


в)



г)

Фиг. 9.1. *Количествени критерии на използване на компресиращи алгоритми при VoIP – а) PCM, б) CS-ACELP, в) MPC-MLQ и г) ACEP*



Фиг. 9.2. *Предимства и недостатъци на VoIP*

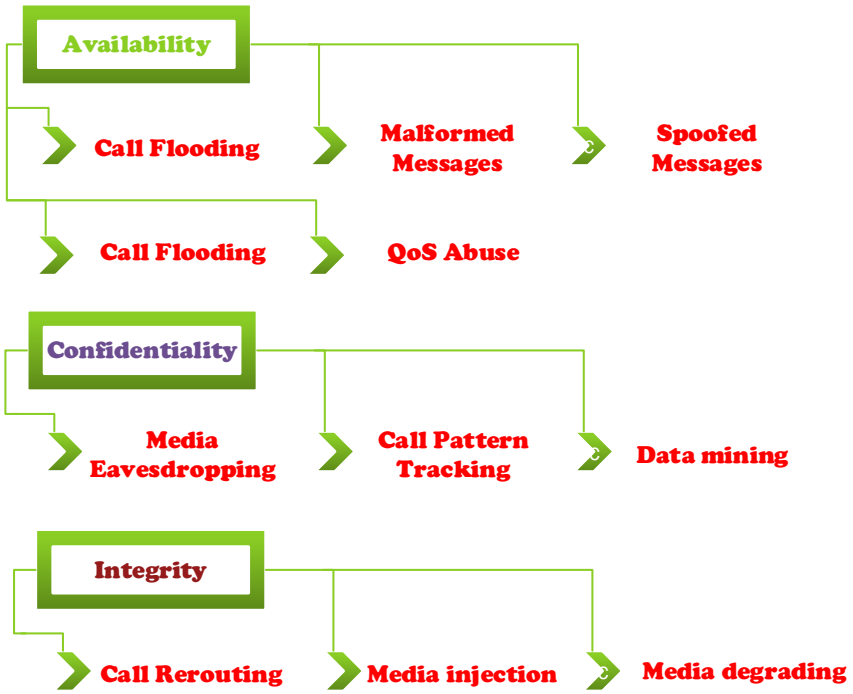
Гласовият и видео трафик притежават известна чувствителност спрямо Packet Loss, Jitter и Delay при пакетно предаване. Ето защо потребителските обслужващи IP мрежи е

необходимо да осигурят съответно ниво на надеждност по отношение на специализирана „честотната“, „латентност“ и т.н. с цел поддръжане на QoS. Появата на „латентност“ и „джитер“ се дължат на ниското качество на Интернет връзката и свързващите Ethernet кабели, използване на неподходящ спрямо изискванията за маршрутизатор и др. Могат да бъдат обобщени следните предимства и недостатъци на VoIP на фиг. 9.2 [17].

Друг съществен аспект се свързва със сигурността на VoIP услугата с цел осуряване на поверителност на провежданите потребителски разговори по трите основни направления:

- *I: Availability;*
- *II: Confidentiality;*
- *III: Integrity.*

Във връзка с това на фиг. 9.3 са посочени типовете атаки спрямо оказаните направления.

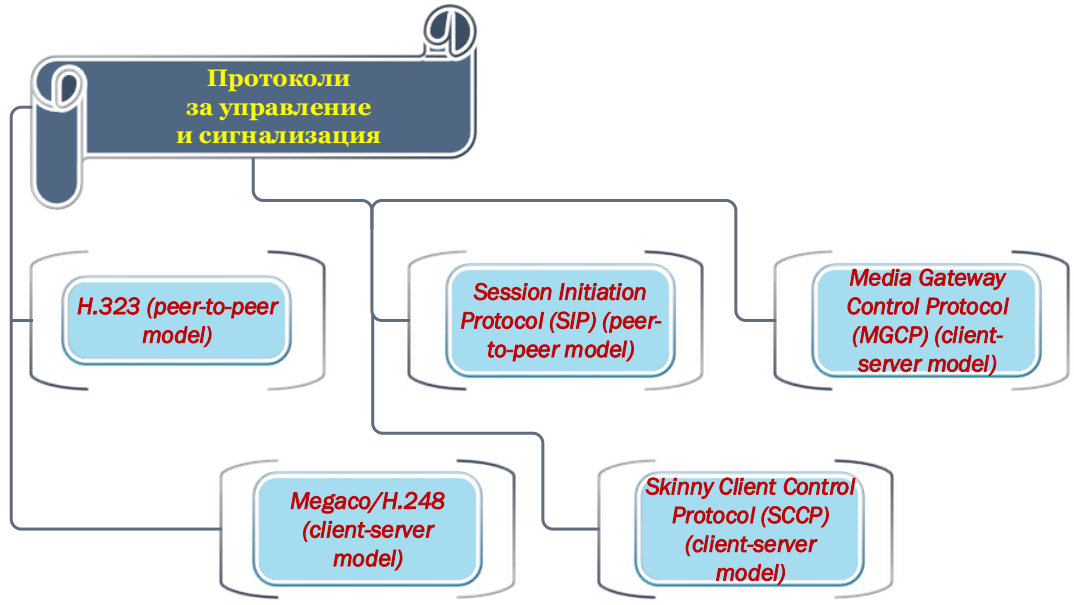


Фиг. 9.3. *Видове атаки по направления Availability, Confidentiality и Integrity*

Мрежовите протоколи, използвани за реалиция и управление на функционалността на VoIP технологията, се разделят в две групи:

- *I група: Протоколи за управление и сигнализация (фиг. 9.4.а);*
- *II група: Протоколи за пренос на гласови данни (фиг. 9.4.б).*

Съответните технически средства са стандартизирани от редица международни организации, сред които могат да бъдат посочени:



а)

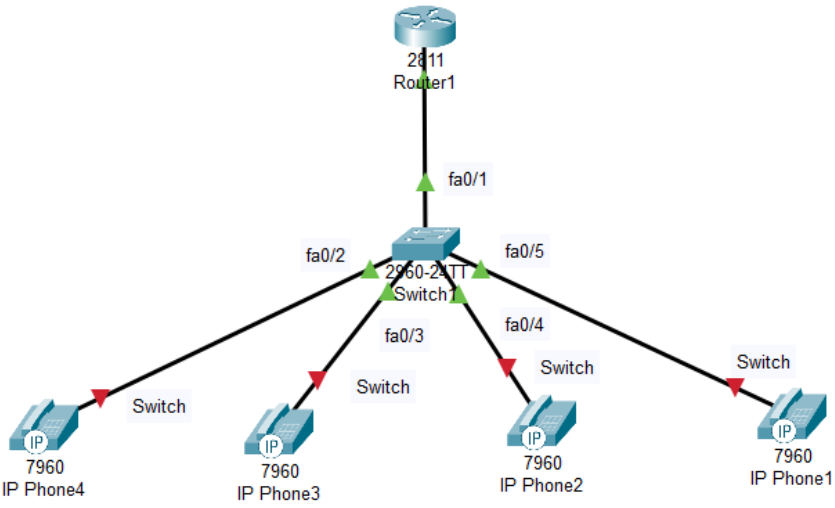
б)

Фиг. 9.4. *Протоколи за реализация на VoIP технологията – а) управление и сигнализация и б) пренос на гласови пакети*

- *Международен телекомуникационен съюз (ITU - International Telecommunication Union) със седалище Женева, Швейцария;*
- *Съюза на Интернет инженерите (IETF - Internet Engineering Task Force) и т.н.* [18].

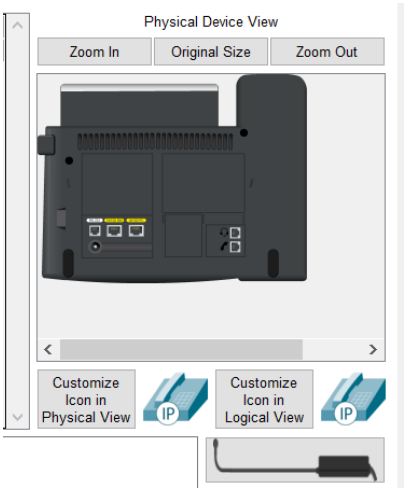
9.3. Последователност на работа и задачи за изпълнение

- Изграждане на тестова мрежова VoIP топология, показана на фиг. 9.5 с включване на маршрутизатор, комутатор и IP телефони;

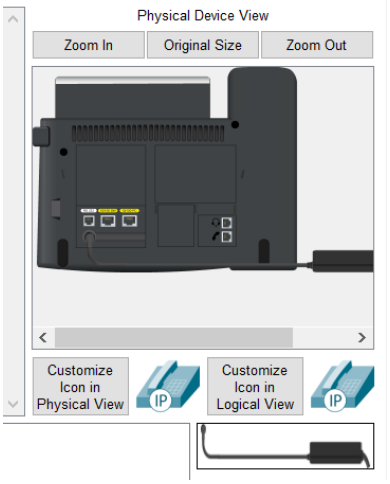


Фиг. 9.5. *Свързване на мрежовата VoIP конфигурация*

- Добавяне на специфициран VoIP захранващ адаптер към обособения за целта конектор според фиг. 9.6, след което следва активация на връзките между мрежовите IP телефони и комутатора на фиг. 9.7;

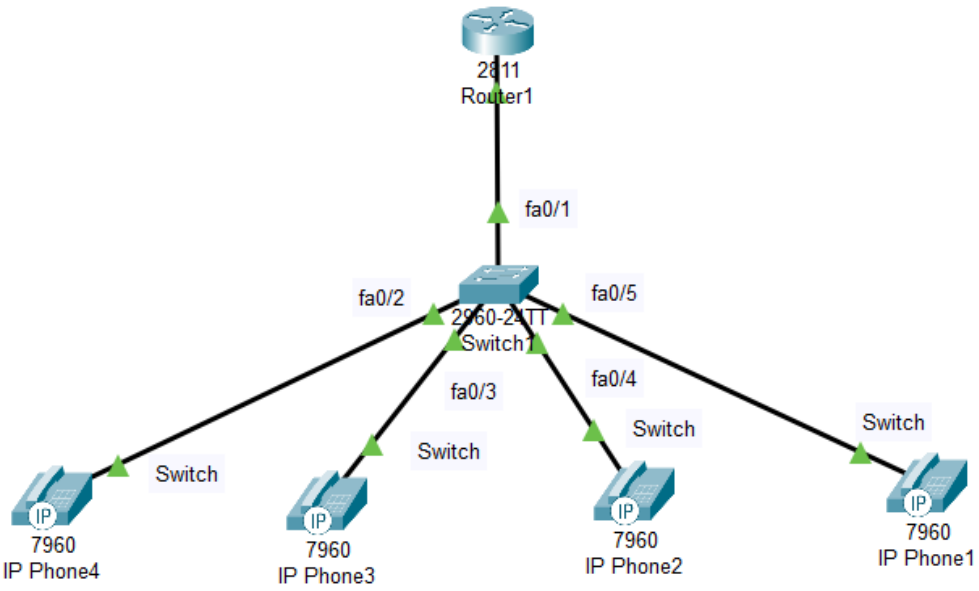


а)



б)

Фиг. 9.6. *Включване на VoIP захранващ адаптер – а) IP телефон а) без и б) с наличие на адаптер*



Фиг. 9.7. *Активиране на портовете от IP Phones към комутатора след включване на VoIP захранващия адаптер*

- Конфигуриране на комуникационния интерфейс FastEthernet 0/0 и разрешаване на опцията за динамично назначаване на IP адреси DHCP на мрежовия маршрутизатор, осъществено на фиг. 9.8;

```
Router>enable
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fa0/0
Router(config-if)#ip address 192.168.10.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
```

Фиг. 9.8. *Настройка на fa0/0 интерфейса на Router1*

- Дефиниране на DHCP обхват по отношение на VLAN структурата с име VOICE. Указване на DHCP мрежа и рутер по подразбиране. Извършване на VoIP конфигурация във връзка с TFTP сървър. Тук се осигурява възможността на всеки мрежови IP телефон да бъде свързан към мрежата посредством съответен IP адрес и локация на TFTP сървър. Процедурата е показана на фиг. 9.9;

```
Router(config-if)#ip dhcp pool VOICE
Router(dhcp-config)#network 192.168.10.0 255.255.255.0
Router(dhcp-config)#default-router 192.168.10.1
Router(dhcp-config)#option 150 ip 192.168.10.1
Router(dhcp-config)#exit
Router(config)#
Router(config)#
```

Фиг. 9.9. *Назначаване на мрежа с DHCP обхват за връзка на мрежовите IP телефони*

- Конфигуриране на управление на обажданията във връзка с предоставяната VoIP телефонната услуга (Call Manager Express Telephony Service) относно мрежовия маршрутизатор. Реализира се фиг. 9.10 чрез дефиниране на:

- ✓ Максималния брой директни обаждания;
- ✓ Максималния брой IP телефонни апарати;
- ✓ Източника на IP пакети във връзка с пренос на глас;
- ✓ Автоматично назначаване на цифри за набиране към бутоните на мрежовите IP телефони.

```
Router>enable
Router#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#telephony-service
Router(config-telephony)#max-dn 5
Router(config-telephony)#max-ephones 5
Router(config-telephony)#ip source-address 192.168.10.1 port 2000
Router(config-telephony)#auto assign 4 to 6
Router(config-telephony)#auto assign 1 to 5
Router(config-telephony)#
```

Фиг. 9.10. *Настройки относно Call Manager Express Telephony Service*

- ➢ Назначаване на унифициращи номера за връзка и реализиране на комуникация по мрежови IP телефони за пренос на глас с помощта на IP пакетна организация в CLI на мрежовия маршрутизатор, показано на фиг. 9.11;

```
Router(config-ephone-dn)#number 54001
Router(config-ephone-dn)#exit
Router(config)#%DHCPD-4-PING_CONFLICT: DHCP address conflict:
server pinged 192.168.10.1.

%IPPHONE-6-REGISTER: ephone-1 IP:192.168.10.3 Socket:2
DeviceType:Phone has registered.
%DHCPD-4-PING_CONFLICT: DHCP address conflict:  server pinged
192.168.10.1.
```

а)

```
Router(config)#ephone-dn 2
Router(config-ephone-dn)#%LINK-3-UPDOWN: Interface ephone_dsp DN
2.1, changed state to up

Router(config-ephone-dn)#number 54002
Router(config-ephone-dn)#exit
Router(config)#
%IPPHONE-6-REGISTER: ephone-3 IP:192.168.10.5 Socket:2
DeviceType:Phone has registered.
```

б)

```
Router(config)#ephone-dn 3
Router(config-ephone-dn)#%LINK-3-UPDOWN: Interface ephone_dsp DN
3.1, changed state to up

Router(config-ephone-dn)#number 54003
Router(config-ephone-dn)#exit
Router(config)#
%IPPHONE-6-REGISTER: ephone-2 IP:192.168.10.6 Socket:2
DeviceType:Phone has registered.
```

в)

```
Router(config)#ephone-dn 4
Router(config-ephone-dn)#%LINK-3-UPDOWN: Interface ephone_dsp DN
4.1, changed state to up

Router(config-ephone-dn)#number 54004
Router(config-ephone-dn)#exit
Router(config)#
%IPPHONE-6-REGISTER: ephone-4 IP:192.168.10.4 Socket:2
DeviceType:Phone has registered.
```

г)

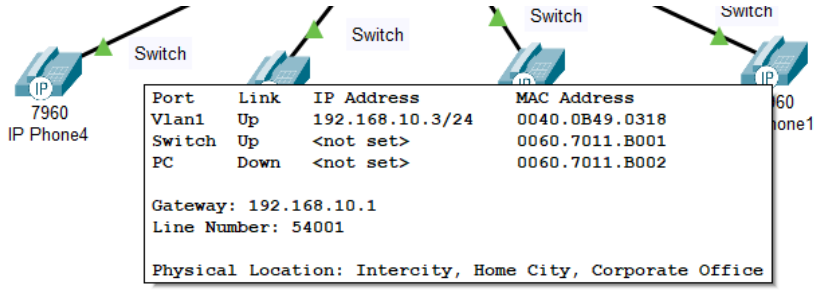
Фиг. 9.11. *Настройка на унифициран номер за връзка и комуникация с мрежово устройство а) IP Phone 1, б) IP Phone 2, в) IP Phone 3 и г) IP Phone 4*

- Настройка на обхвата от портове за достъп – fa0/1 до fa0/5 интерфейси, при мрежовия комутатор, както и конфигурация на разделянето на „гласови“ и „информационни“ тип данни в различни VLAN структури (фиг. 9.12). Информационните пакети ще бъдат достъпни посредством VLAN;

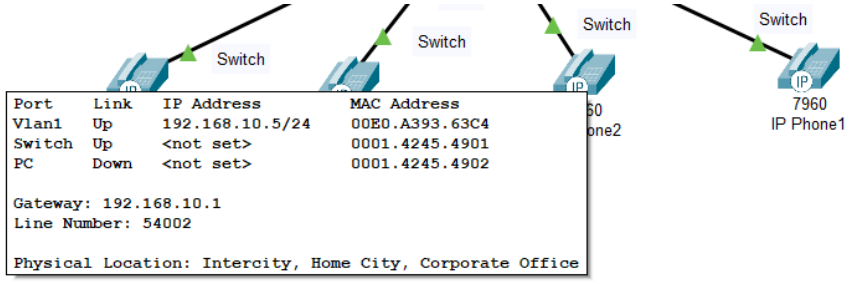
```
Switch>enable
Switch#config term
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#interface range fa0/1-5
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport voice vlan 1
Switch(config-if-range)#exit
Switch(config)#
```

Фиг. 9.12. *Конфигуриране на VLAN VOICE за пренос на глас*

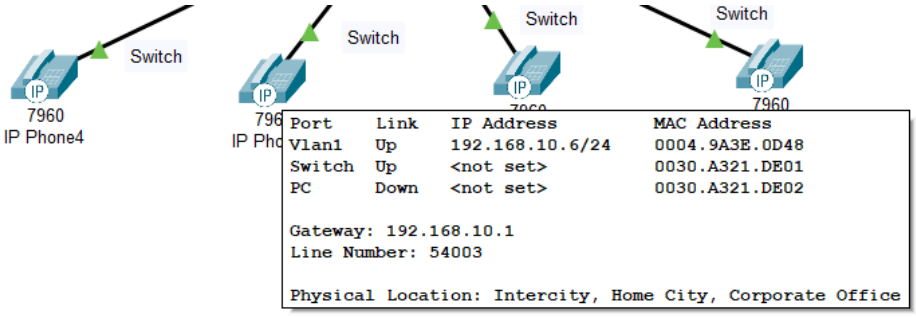
- Верификация относно коректността на назначаване на унифициращи номера за връзка и реализиране на комуникация по отношение на мрежови IP телефони за пренос на гласови данни, показана на фиг. 9.13;



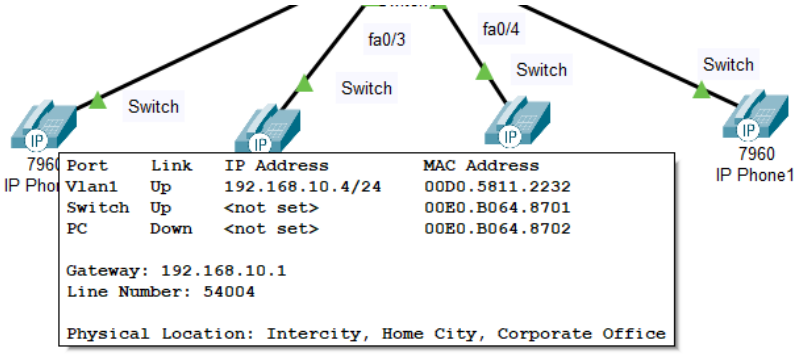
а)



б)

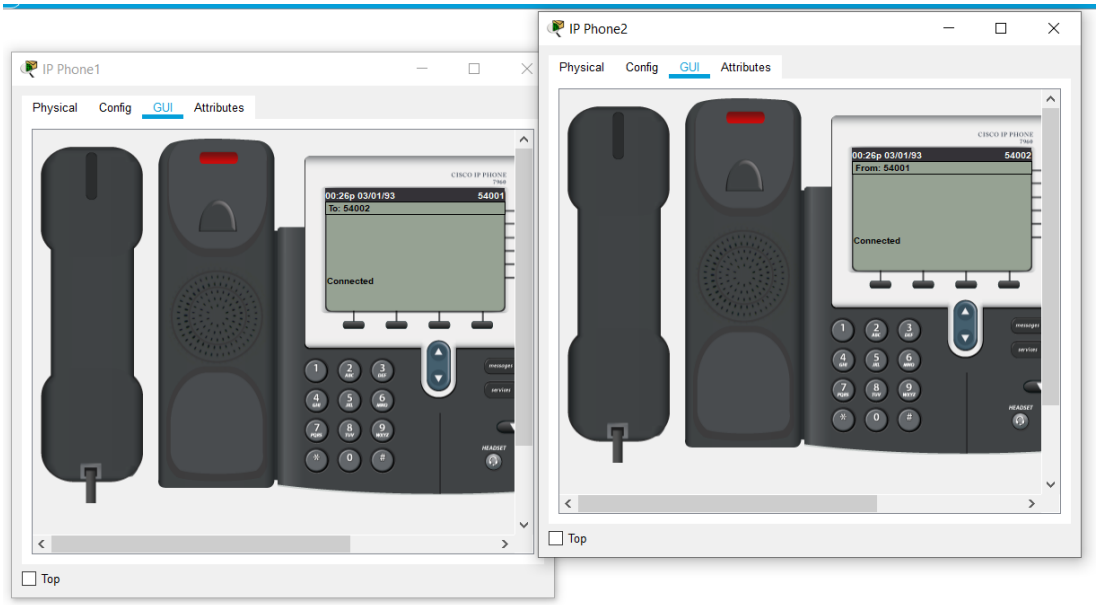


в)

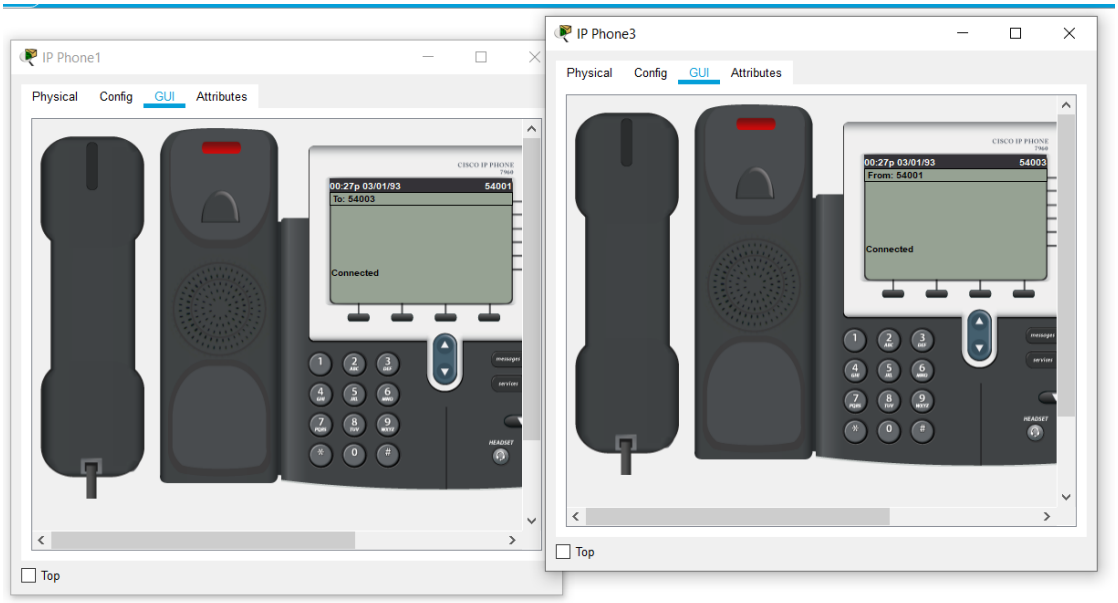


г)

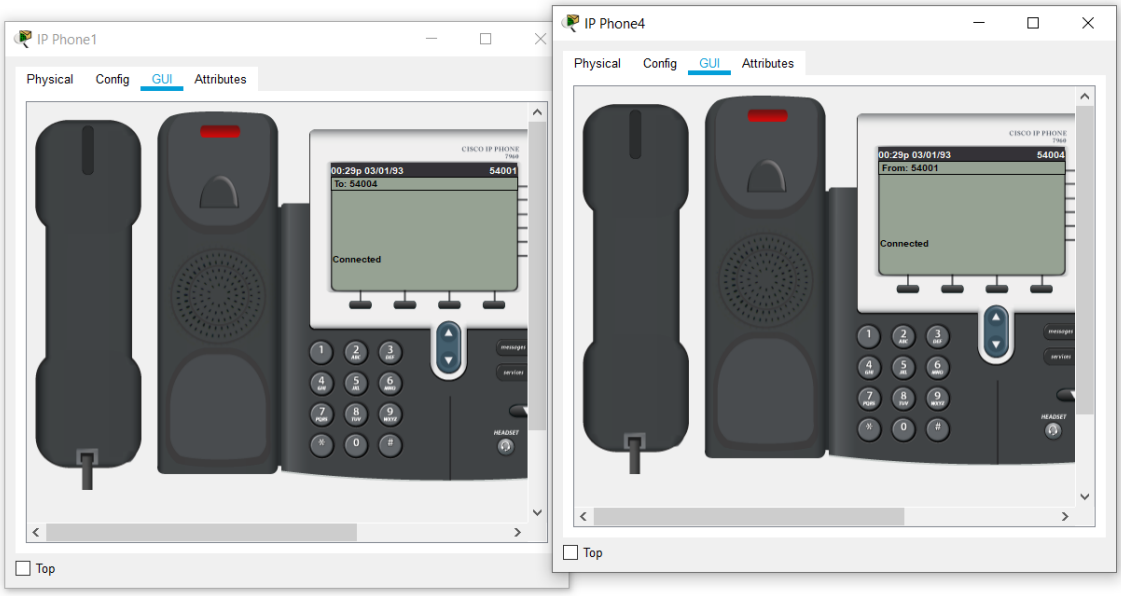
Фиг. 9.13. *Верификация на коретността на задаване на унифициран номер за връзка и комуникация с мрежово устройство а) IP Phone 1, б) IP Phone 2, в) IP Phone 3 и г) IP Phone 4*



а)



б)



в)

Фиг. 9.14. *Верификация на коретността на свързаност между мрежовите IP телефони – а) от IP Phone 1 към IP Phone 2, б) от IP Phone 1 към IP Phone 3 и в) от IP Phone 1 към IP Phone 4*

- Тестване на VoIP комуникацията относно свързаност и коректност на предаването на гласови данни между мрежовите IP телефони, както е показано на фиг. 9.14.

9.4. Контролни въпроси.

1. Кои са основните критерии за сравнение между прилаганите компресиращи алгоритми при VoIP технологията?
2. Какви са предимствата и недостатъците при използване на VoIP?
3. Кои са изискваните приоритети при обезпечаване на повирителността при провеждане на разговори чрез VoIP технологията?
4. Дайте примери за видове атаки съобразно основополагащите направления, свързани със сигурността при VoIP.
5. Кои са поддържаните стандарти за реализация и управление на функционалността при VoIP?

Лабораторно упражнение № 10

ИЗГРАЖДАНЕ, КОНФИГУРИРАНЕ И УПРАВЛЕНИЕ НА ТРАФИКА В VLAN МРЕЖОВИ ИНФРАСТРУКТУРИ ЧРЕЗ ПОМОЩТА НА CISCO PACKET TRACER

10.1. Цел на упражнението

Да се запознаят студентите с последователността на настройка и конфигурация на VLAN мрежови конфигурация като подход за обезпечаване на сигурността и пакетното предаване на данни между осособени мрежови сегменти към потребителски роли и права за достъп на йерархични нива.

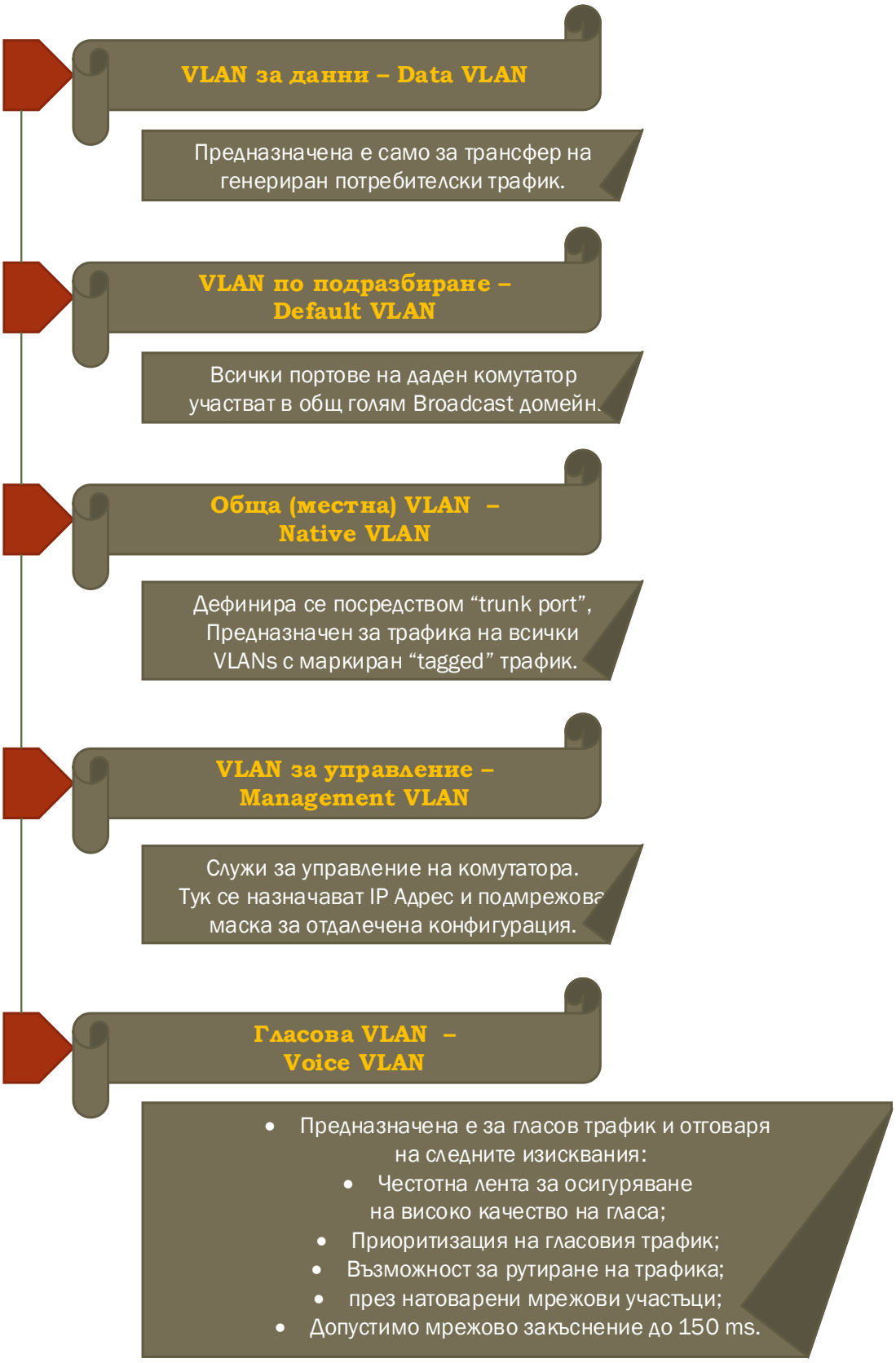
10.2. Теоретична постановка

Виртуалните локални мрежи ли VLAN се разглеждат като метод, чрез който дадена физическа компютърна мрежа се разделя на различни виртуални мрежови сегменти. Този подход се свързва с обезпечаване на логическата организация, сигурността и контрол на достъпа според оторизация на потребителските роли до специфицирани потребителски данни на различни нива. Особеност на VLANs е способността на потребителите да могат да комуникират само в обхвата на своята, но не в с другите виртуални мрежи [19].

Системният мрежови администратор има способността да дефинира логически групи от мрежови устройства, функциониращи като структурна независима мрежа или обособена IP подмрежова структура. По този начин множество от IP мрежи и мрежови сегменти могат да функционират в единна мрежова инфомационно-комуникационна инфраструктура.

Порт базираните VLANS или “port-based VLANs” са едни от най-разпространените, при които конкретен порт на мрежови комутатор се обвързва с дадена VLAN. Някои от основните типове Виртуални локални мрежи са показани на фиг. 10.1. При процедурното дефиниране на VLAN се използват специфицирани идентификатори като се различават няколко режима на определяне на принадлежност на конкретен switch порт към дадена VLAN структура (фиг. 10.2).

С помощта на VLAN се постигат по-висока степен на мрежова гъвкавост, адаптивност и управление на потребителските приложения независимо от физическото разположение и разстоянието между потребителите. Главните предимства от интеграция на VLANs при внедряването на механизми за подобряване на сигурността на трафика са обобщени на фиг. 10.3 [20].



Фиг. 10.1. *Типове Виртуални локални мрежи*

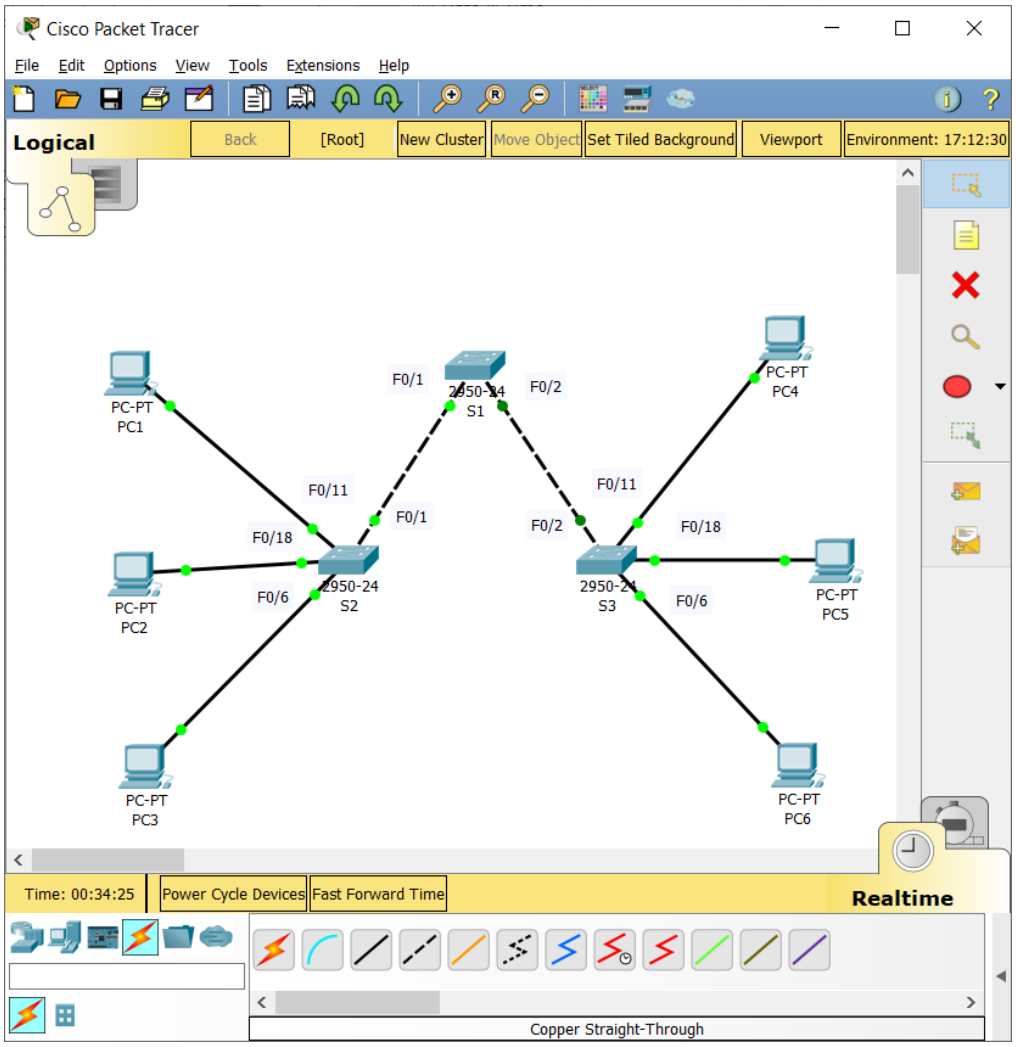
Фиг. 10.2. *Режими на обвързване на switch портове към VLANs*

Фиг. 10.3. *Предимства на VLAN структурите*

10.3. Последователност на работа и задачи за изпълнение

Разглежда се задачата за проектиране и конфигурация на мрежова архитектура с обособени VLANs. Изграждането на топологията ще бъде извършено в среда на продукта Cisco Packet Tracer. Могат да бъдат използвани мрежови устройства „switch“ модели 2900 или 2950. Процедурите включват следдните основни стъпки:

- ❖ Стъпка №1: Създаване на мрежовата топология в работното пространство на симулатора, показана на фиг. 10.4;



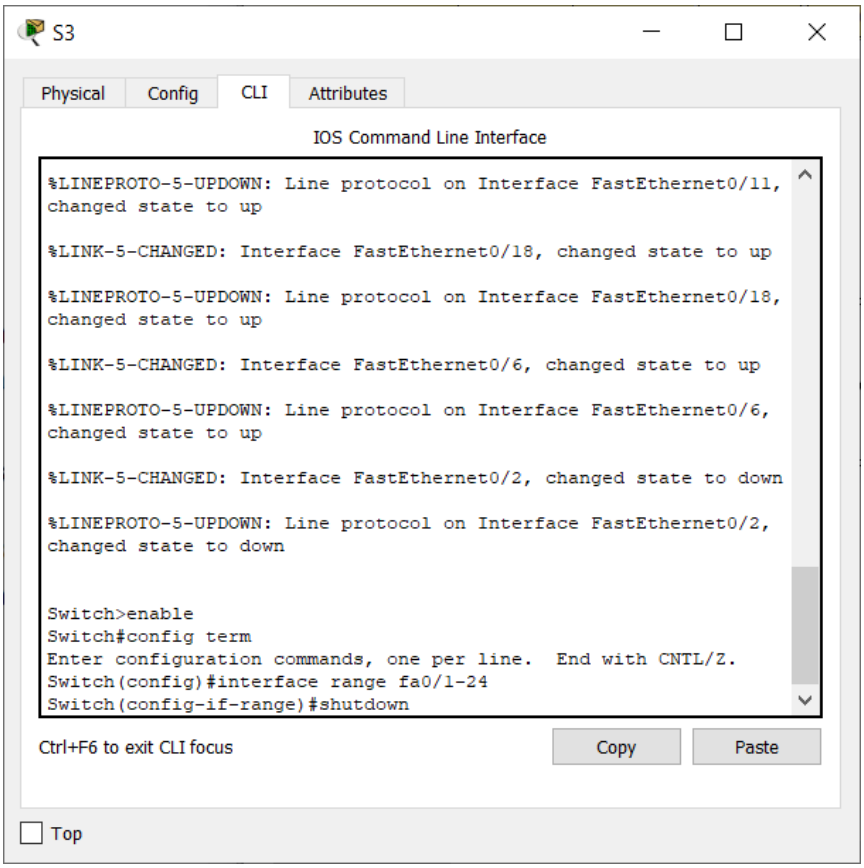
Фиг. 10.4. *VLAN мрежова топология*

Таблица 10.1. Адресна таблица относно конфигурация на PC устройства

Персонална станция	IP адрес	Subnet Mask	Default Gateway
PC1	172.17.10.21	255.255.0.0	172.17.10.1
PC2	172.17.20.22	255.255.0.0	172.17.20.2
PC3	172.17.30.23	255.255.0.0	172.17.30.3
PC4	172.17.10.24	255.255.0.0	172.17.10.4
PC5	172.17.20.25	255.255.0.0	172.17.20.5
PC6	172.17.30.26	255.255.0.0	172.17.30.6

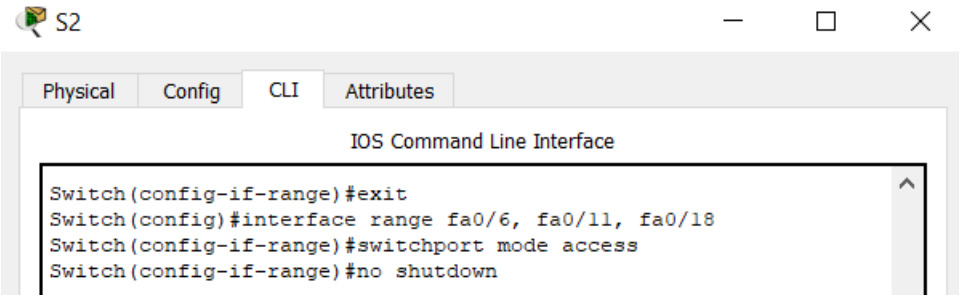
- Стъпка №2: Конфигуриране на IP адреси, подмрежова маска и Gateway по подразбиране за персонални станции от PC0 до PC6, съгласно адресната таблица 10.1;

- Стъпка №3: Изчистване на всички базови конфигурационни настройки и привеждане на всички портове от fa0/1 до fa0/24 в състояние „shutdown“. Стъпката се прилага за всички суичове чрез поредица от команди, въвеждани в CLI – Command Line Interface. Процесът е илюстриран на фиг. 10.5 за S3, а за останалите суичове се изпършва по същия начин;

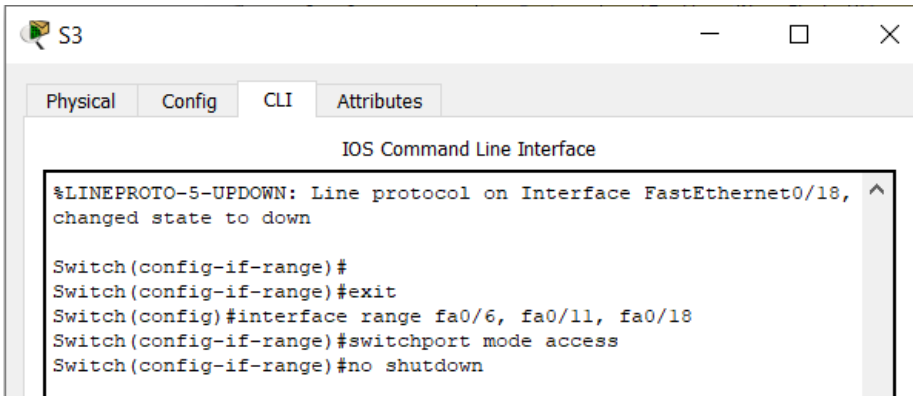


Фиг. 10.5. *Инициализация на портове от fa0/1 до fa0/24 за S3 в състояние „shutdown“*

- Стъпка №4: Разрешаване на портове fa0/6, fa0/11 и fa0/18, към които са свързани персоналните компютърни станции по отношение на мрежовите устройства S2 и S3 – фиг. 10.6;



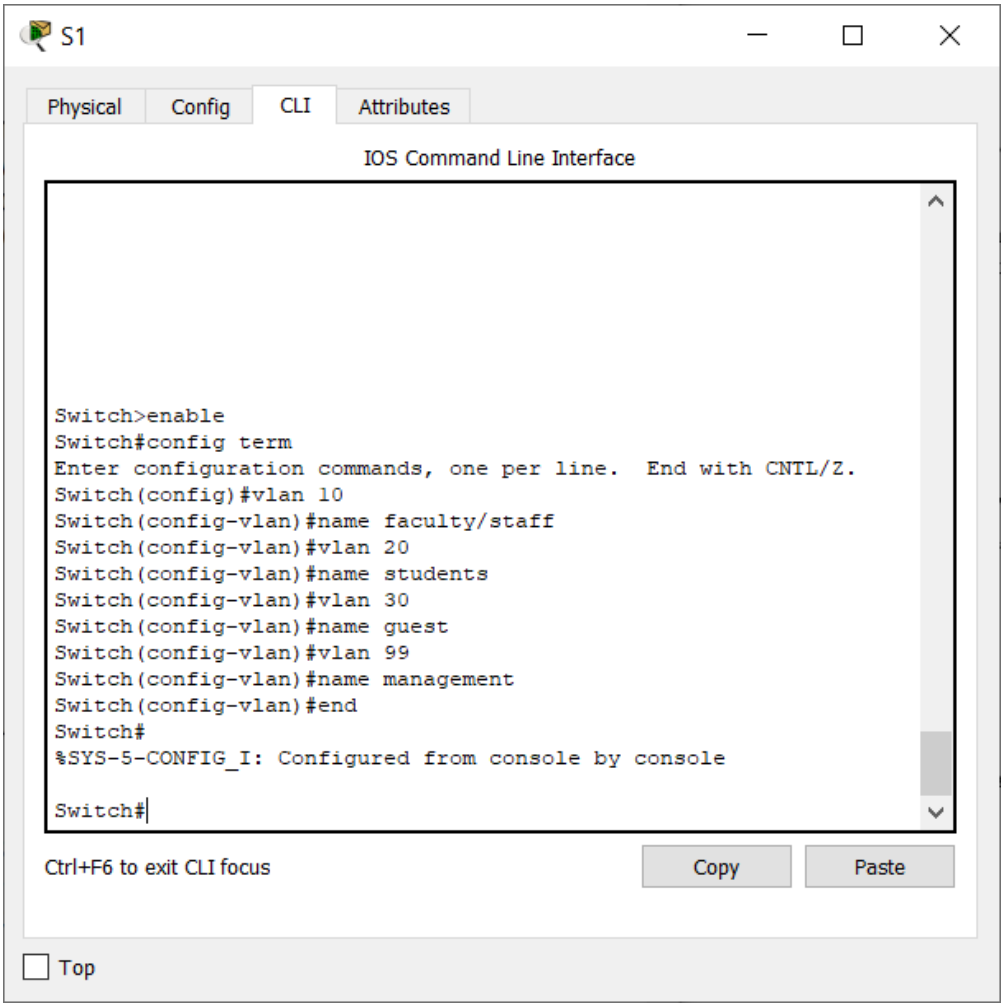
a)



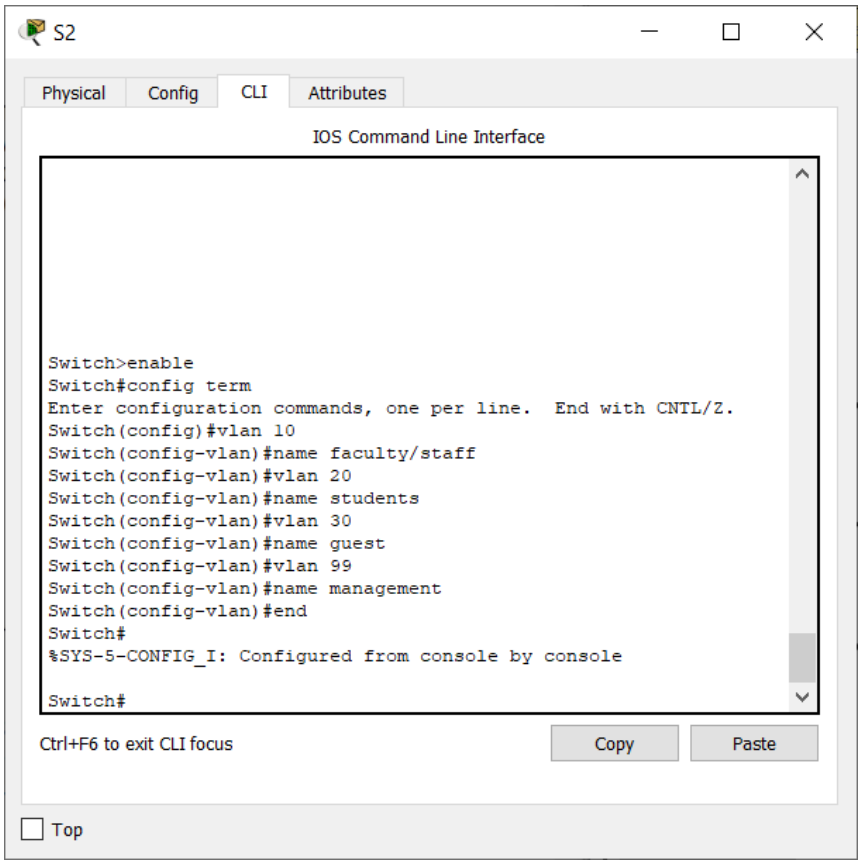
б)

Фиг. 10.6. *Разрешаване на портове от fa0/6, fa0/11 и fa0/18 за а) S2 и б) S3*

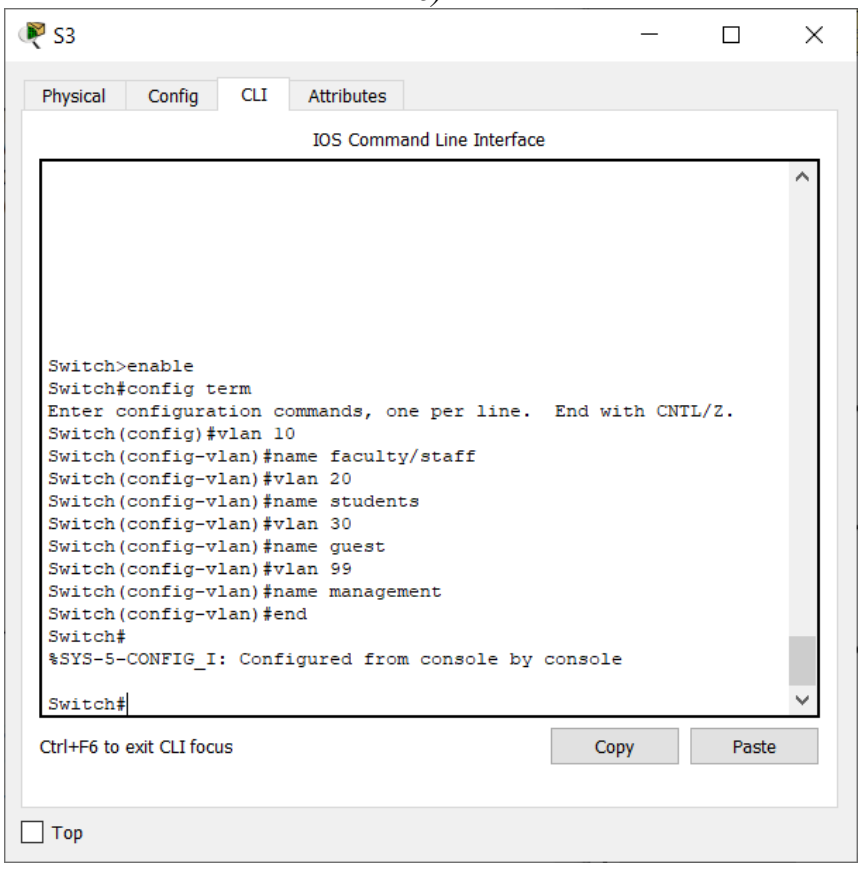
- Стъпка №5: Създаване и верификация на VLANs – VLAN 10 “faculty/staff”, VLAN 20 “students”, VLAN 30 “guest” и VLAN 99 “management”, на мрежовите суичове, S1, S2 и S3. Дейностите са показани на фиг. 10.7 и фиг. 10.8;



а)

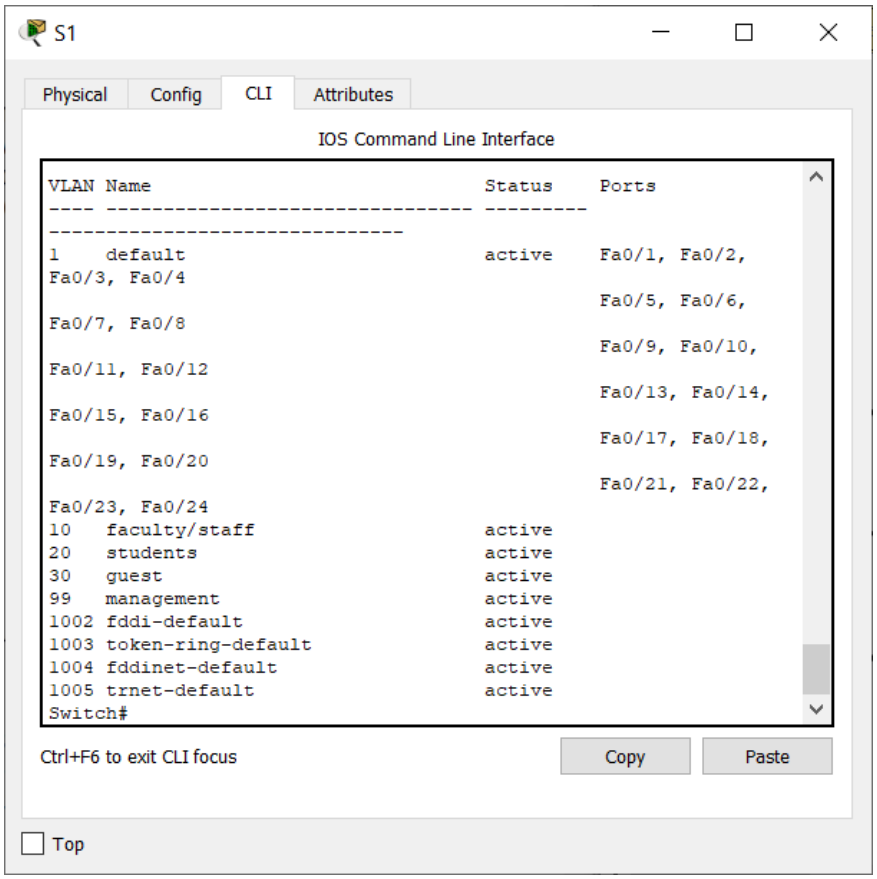


б)

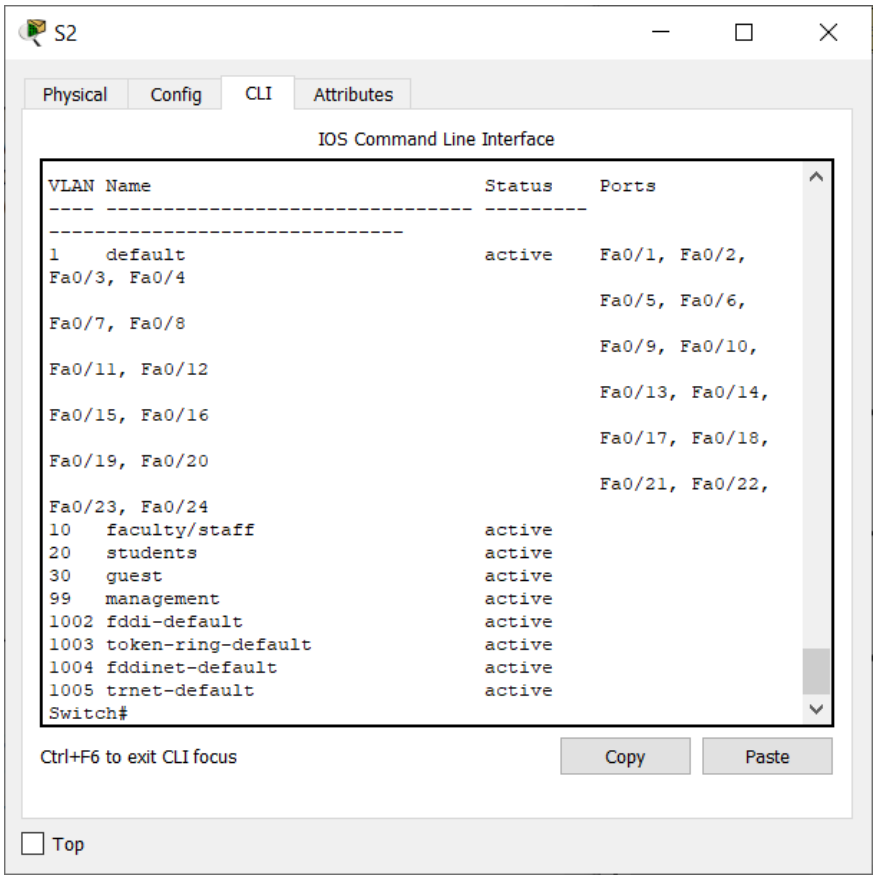


в)

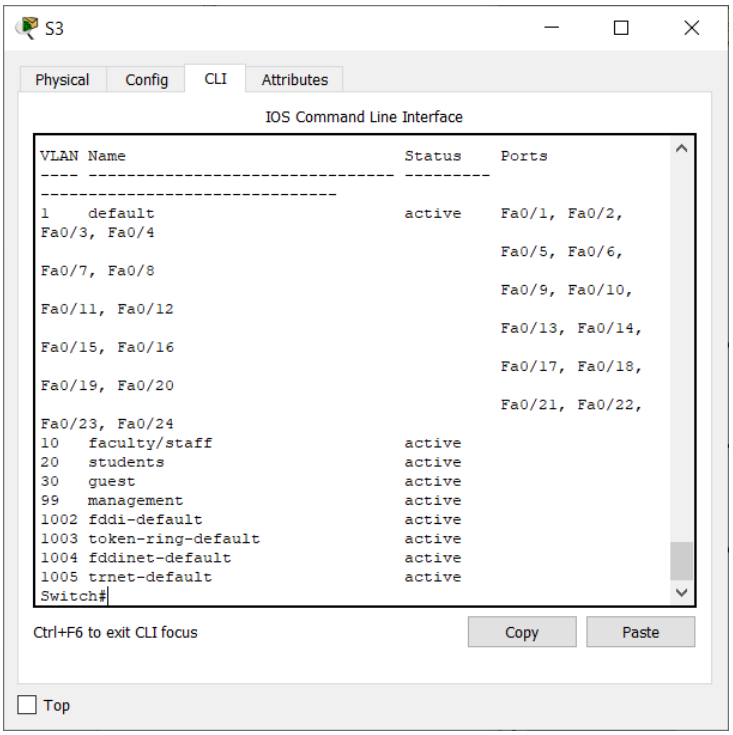
Фиг. 10.7. *Създаване на VLAN 10 "faculty/staff", VLAN 20 "students", VLAN 30 "guest" и VLAN 99 "management" за а) S1, б) S2 и в) S3*


S1
Physical
Config
CLI
Attributes
IOS Command Line Interface
VLAN Name Status Ports
---- -------------------------------- ---------
1 default active Fa0/1, Fa0/2,
Fa0/3, Fa0/4
Fa0/5, Fa0/6,
Fa0/7, Fa0/8
Fa0/9, Fa0/10,
Fa0/11, Fa0/12
Fa0/13, Fa0/14,
Fa0/15, Fa0/16
Fa0/17, Fa0/18,
Fa0/19, Fa0/20
Fa0/21, Fa0/22,
Fa0/23, Fa0/24
10 faculty/staff active
20 students active
30 guest active
99 management active
1002 fddi-default active
1003 token-ring-default active
1004 fddinet-default active
1005 trnet-default active
Switch#
Ctrl+F6 to exit CLI focus
Copy
Paste
Top

а)


S2
Physical
Config
CLI
Attributes
IOS Command Line Interface
VLAN Name Status Ports
---- -------------------------------- ---------
1 default active Fa0/1, Fa0/2,
Fa0/3, Fa0/4
Fa0/5, Fa0/6,
Fa0/7, Fa0/8
Fa0/9, Fa0/10,
Fa0/11, Fa0/12
Fa0/13, Fa0/14,
Fa0/15, Fa0/16
Fa0/17, Fa0/18,
Fa0/19, Fa0/20
Fa0/21, Fa0/22,
Fa0/23, Fa0/24
10 faculty/staff active
20 students active
30 guest active
99 management active
1002 fddi-default active
1003 token-ring-default active
1004 fddinet-default active
1005 trnet-default active
Switch#
Ctrl+F6 to exit CLI focus
Copy
Paste
Top

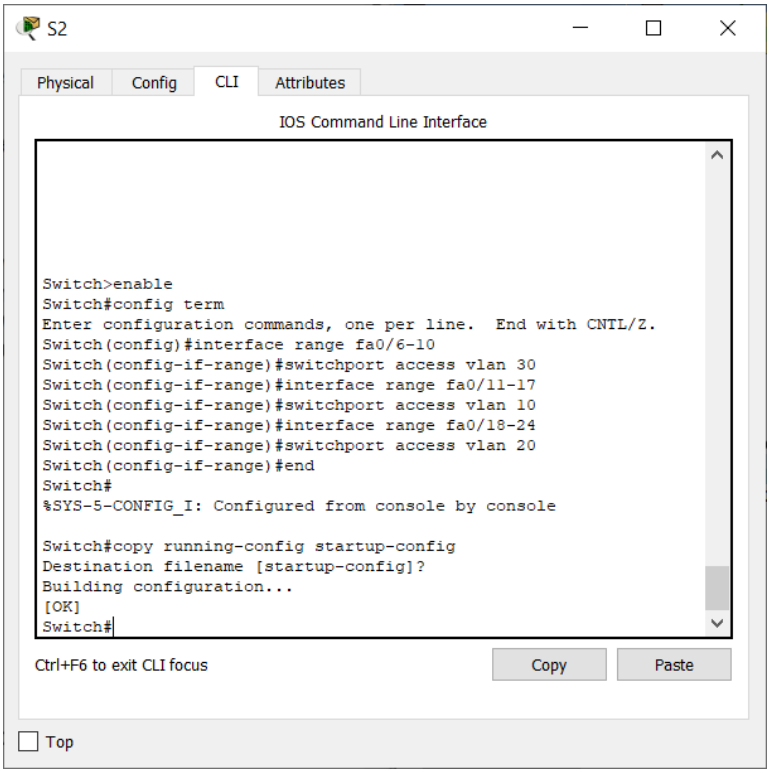
б)



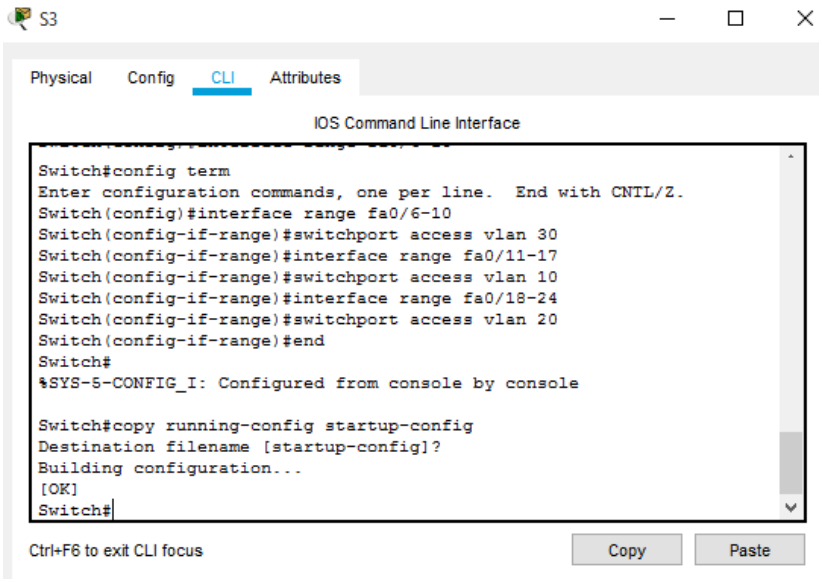
в)

Фиг. 10.8. *Верификация относно коректността на създадените мрежи VLAN 10, VLAN 20, VLAN 30 и VLAN 99 "management" за а) S1, б) S2 и в) S3*

- Стъпка №6: Назначаване на обхвати от интерфейсни портове в рамките на създадените виртуални локални мрежи, съдържащи мрежови устройства S2 и S3. С други думи тук се извършва на назначаване на портове към VLANs в интерфейсен конфигурационен режим (фиг. 10.9);



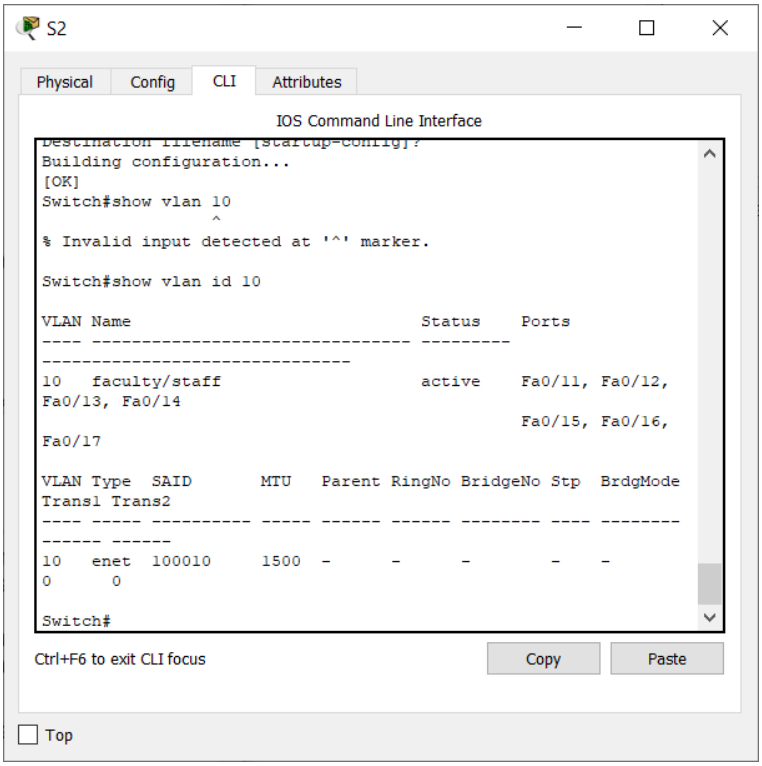
а)



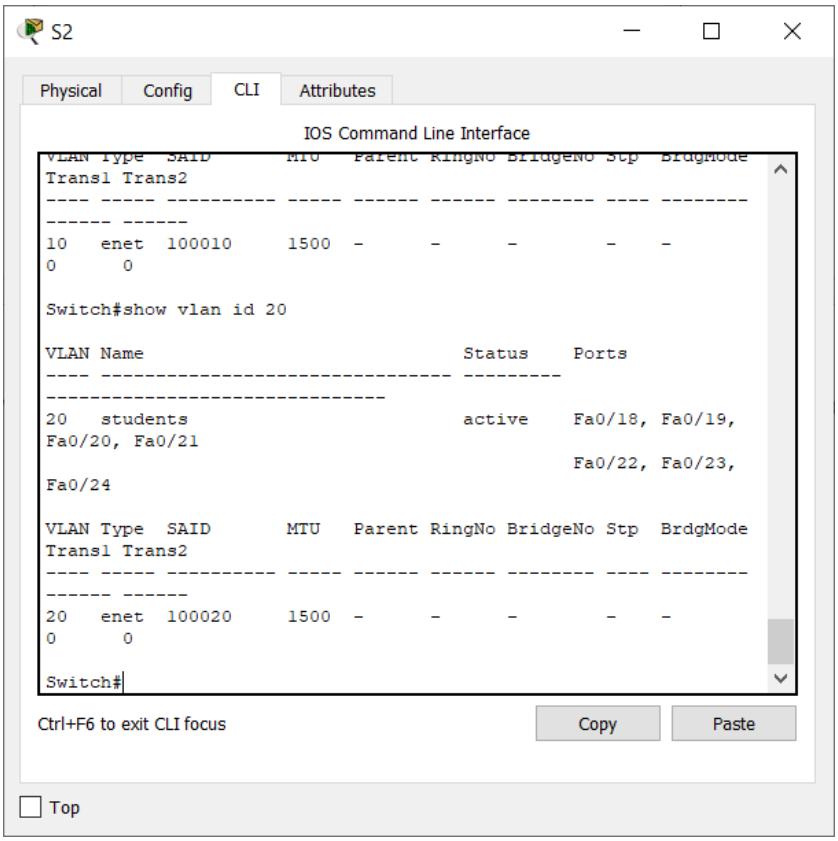
б)

Фиг. 10.9. *Назначаване на портове към VLANs в интерфейсен конфигурационен режим на а) S2 и б) S3*

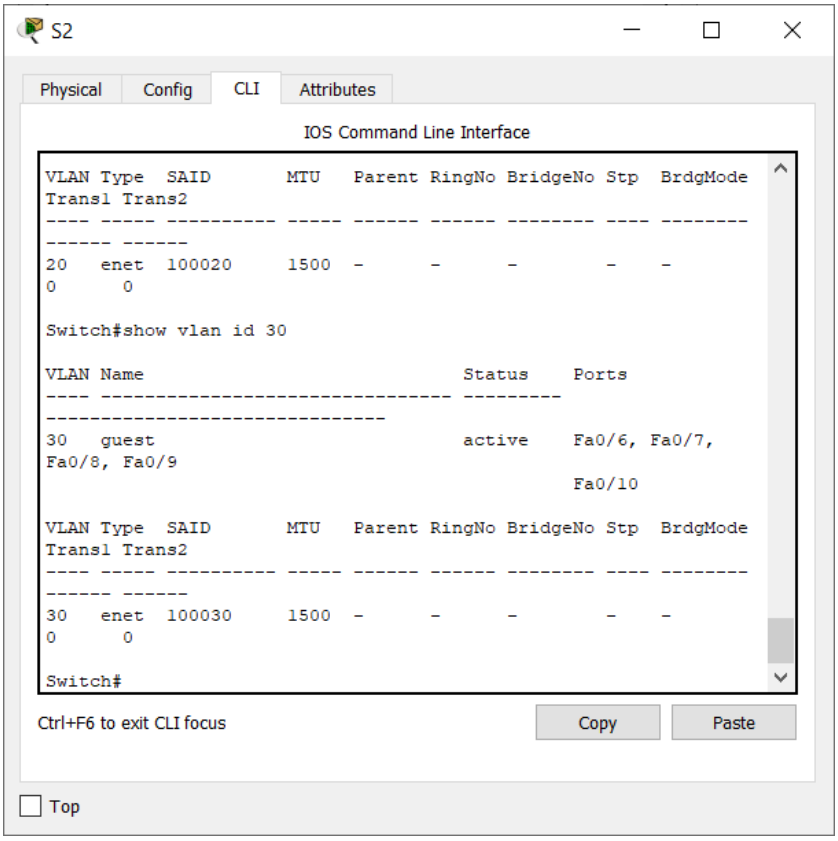
- Стъпка №7: Извършване на проверка относно добавените интрефейсни портове към виртуалните локални мрежи, представена на фиг. 10.10 за суича S2 и фиг. 10.11 за мрежовото устройство S3;



а)

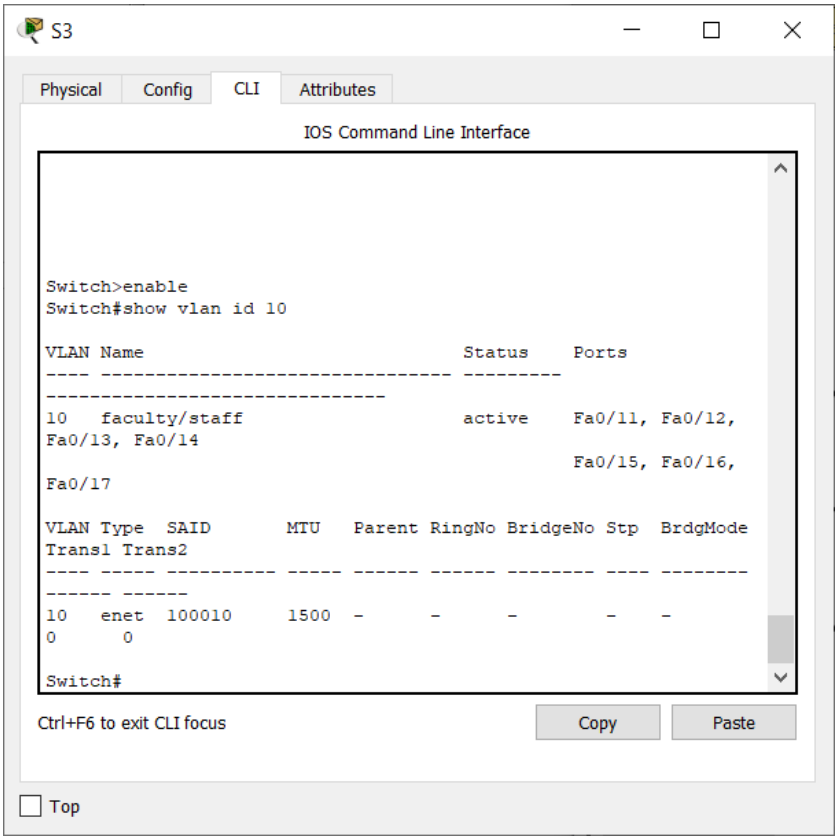


б)

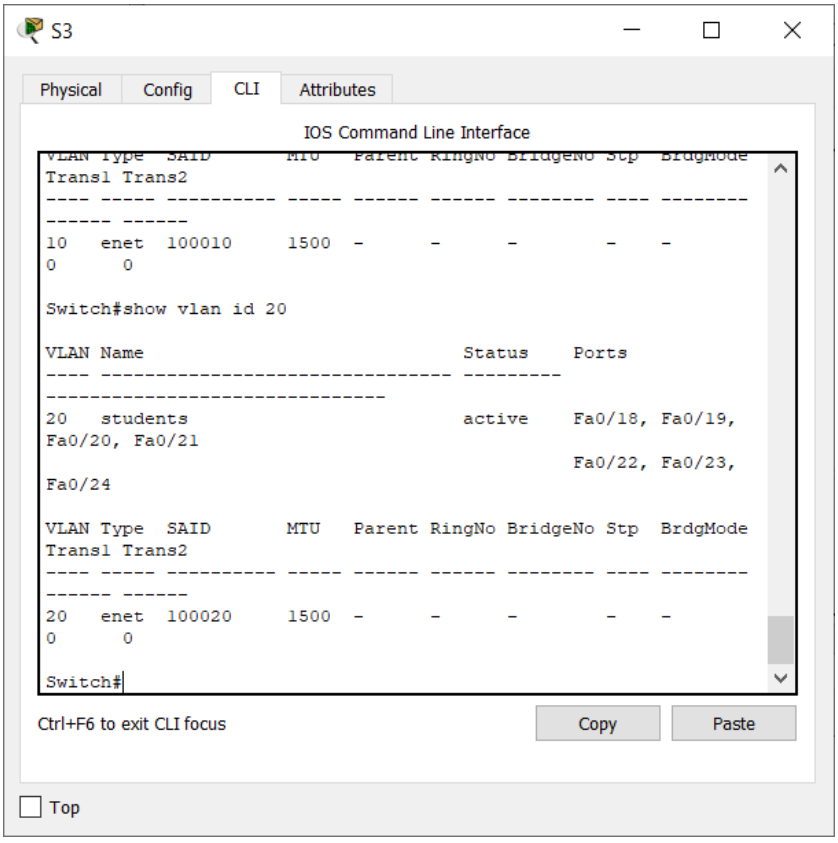


в)

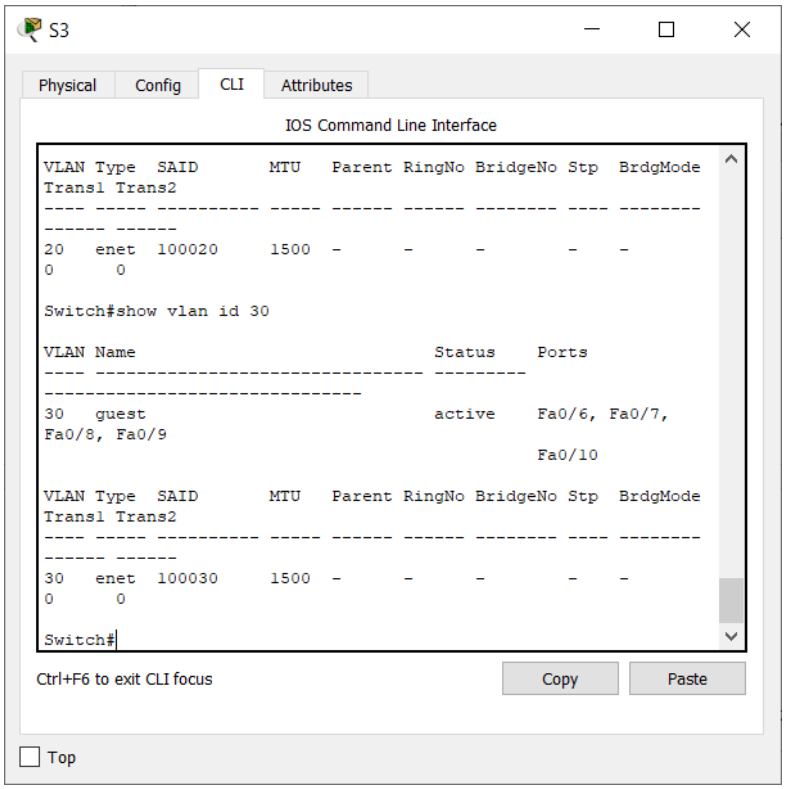
Фиг. 10.10. *Проверка относно добавени портове при S2 за а) VLAN 10, б) VLAN 20 и в) VLAN 30*


S3
Physical
Config
CLI
Attributes
IOS Command Line Interface
Switch>enable
Switch#show vlan id 10
VLAN Name Status Ports
10 faculty/staff active Fa0/11, Fa0/12,
Fa0/13, Fa0/14
Fa0/15, Fa0/16,
Fa0/17
VLAN Type SAID MTU Parent RingNo BridgeNo Stp BrdgMode
Trans1 Trans2
10 enet 100010 1500 - - - - -
0 0
Switch#
Ctrl+F6 to exit CLI focus
Copy
Paste
Top

а)


S3
Physical
Config
CLI
Attributes
IOS Command Line Interface
Trans1 Trans2
10 enet 100010 1500 - - - - -
0 0
Switch#show vlan id 20
VLAN Name Status Ports
20 students active Fa0/18, Fa0/19,
Fa0/20, Fa0/21
Fa0/22, Fa0/23,
Fa0/24
VLAN Type SAID MTU Parent RingNo BridgeNo Stp BrdgMode
Trans1 Trans2
20 enet 100020 1500 - - - - -
0 0
Switch#
Ctrl+F6 to exit CLI focus
Copy
Paste
Top

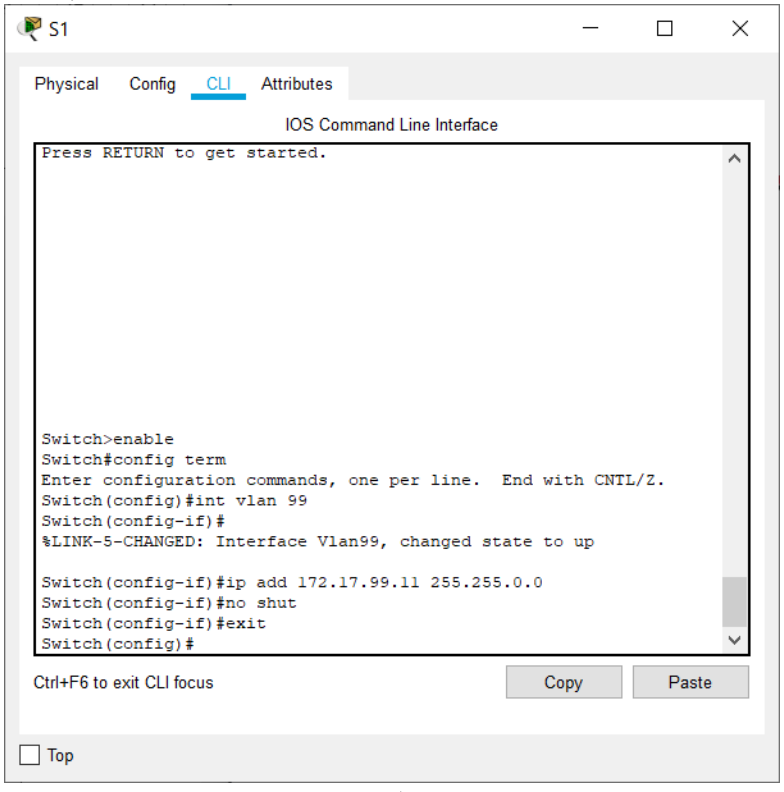
в)



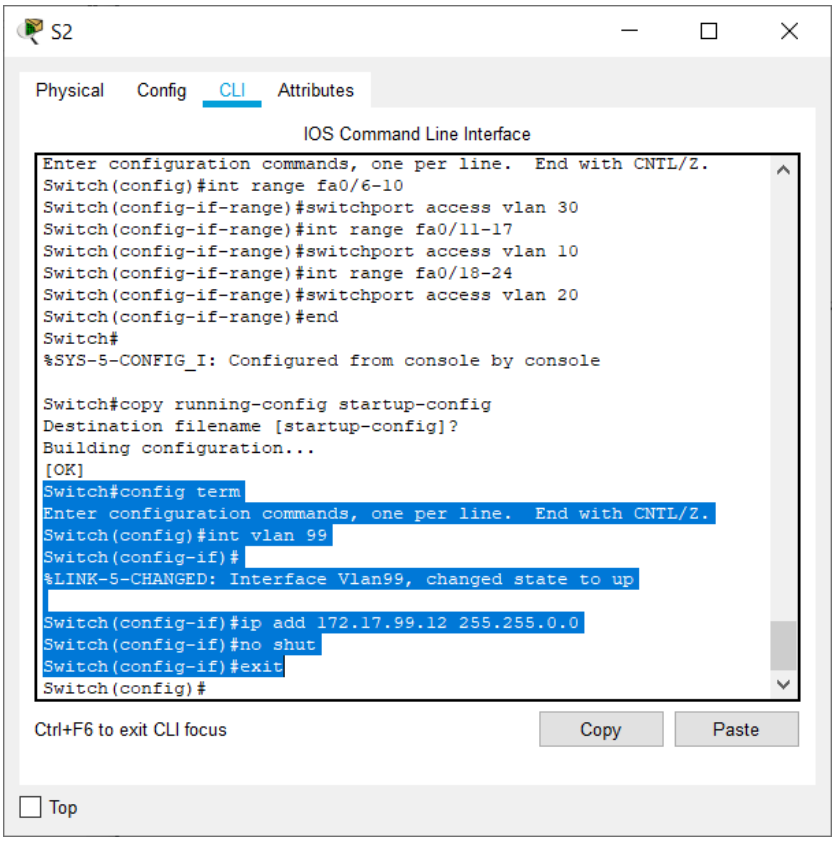
в)

Фиг. 10.11. *Проверка относно добавени портове при S3 за а) VLAN 10, б) VLAN 20 и в) VLAN 30*

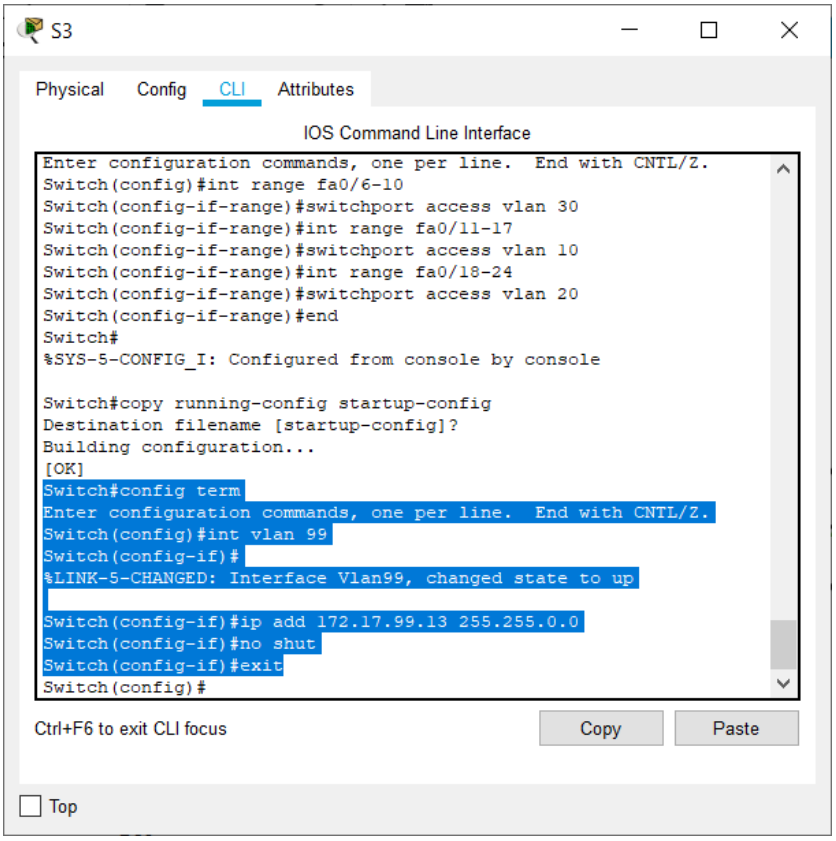
- Стъпка №8: Назначаване и конфигуриране на VLAN 99 “management” - IP адрес и Subnet Mask, последователно за S1, S2 и S3, както е илюстрирано на фиг. 10.12;



а)



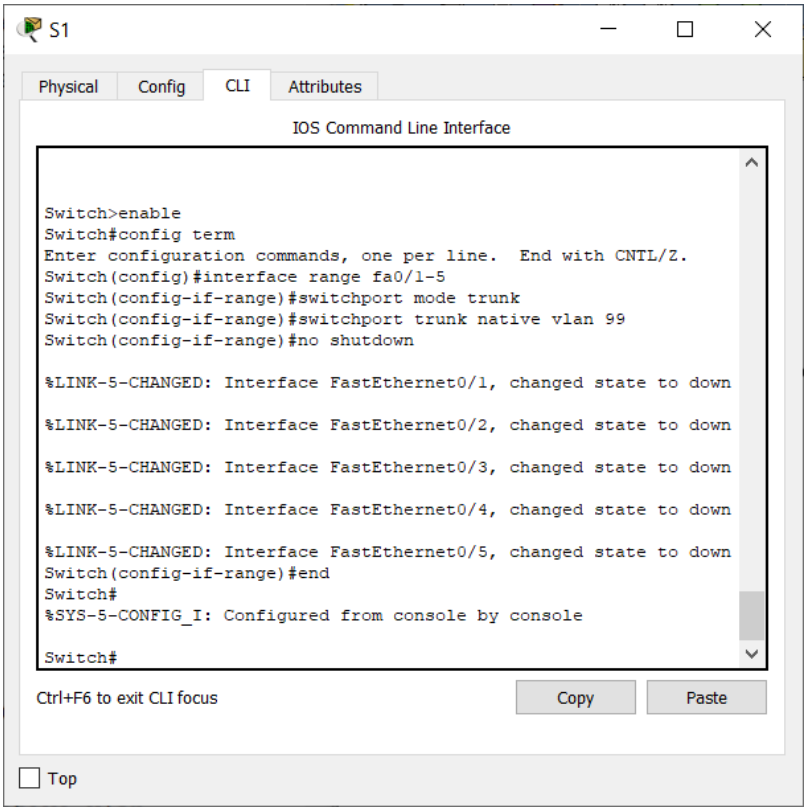
б)



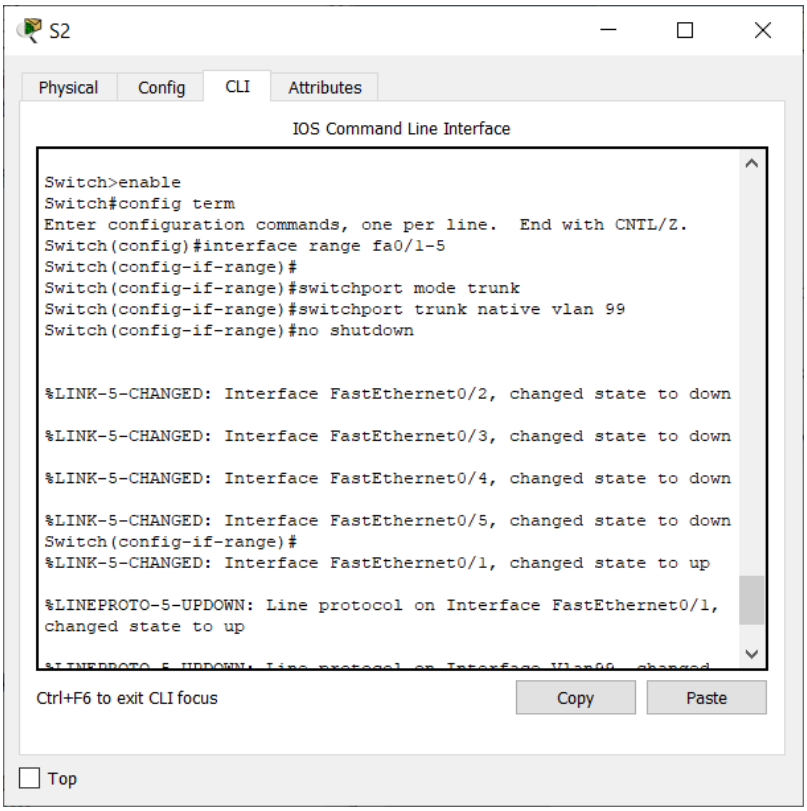
в)

Фиг. 10.12. *Назначаване на IP адрес и подмрежова маска за VLAN 99 при а) S1, б) S2 и в) S3*

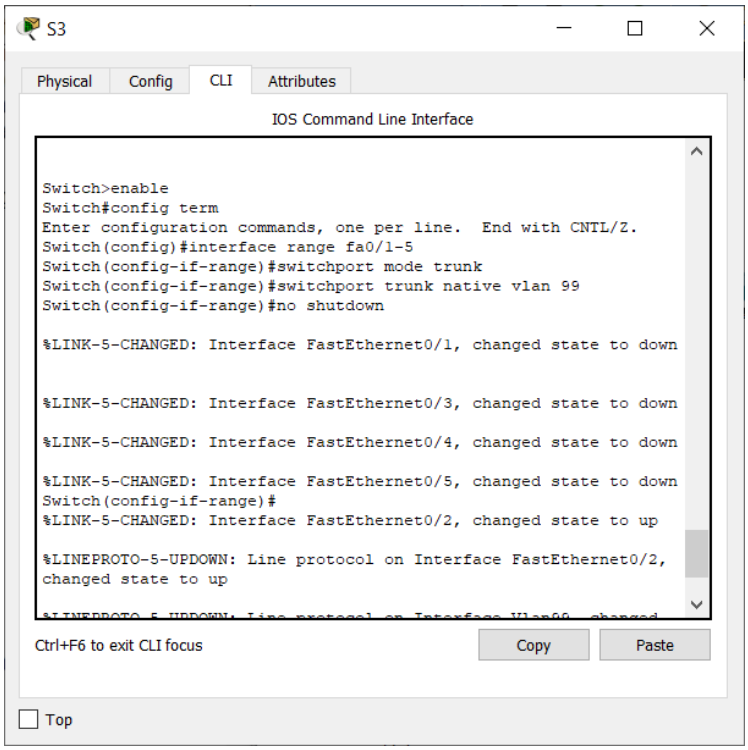
❖ Стъпка №9: Конфигуриране на информационни канали за връзка за портовете на всички мрежови устройства “Switch”. Процесите са реализирани чрез набор от инструкции, въведени в CLI на фиг. 10.13;



а)



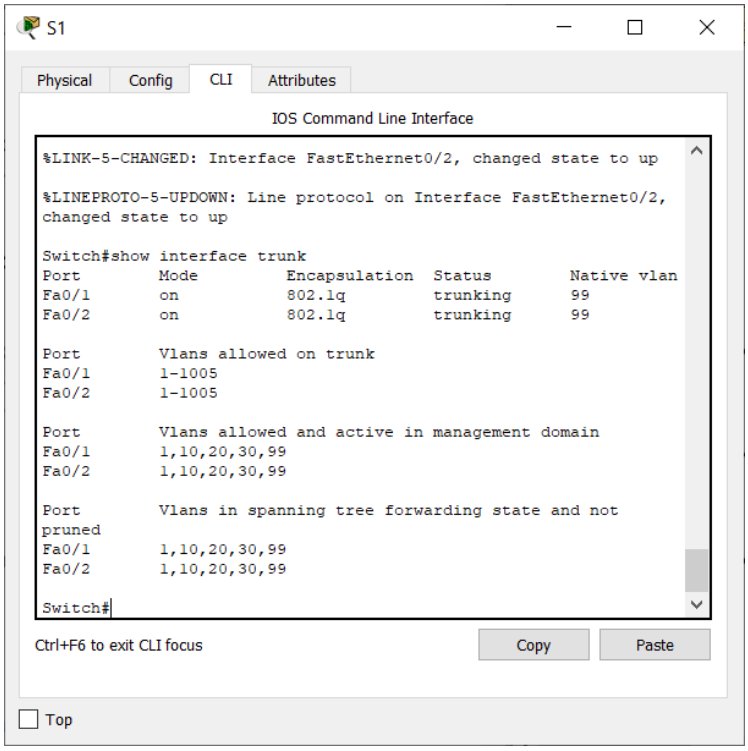
б)



в)

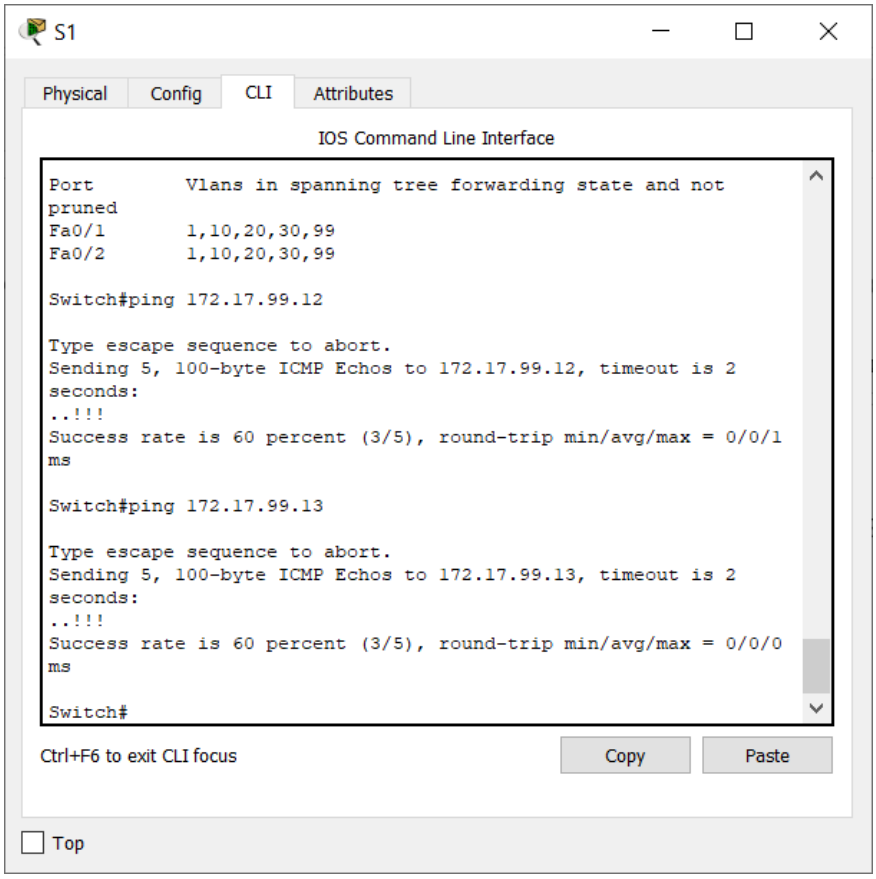
Фиг. 10.13. *Конфигуриране на информационни канали за връзка за избраните обхвати от портовете по отношение на а) S1, б) S2 и в) S3*

❖ Стъпка №10: Провеждане на верификация относно конфигурираните информационни канали за връзка при S1, показана на фиг. 10.14;

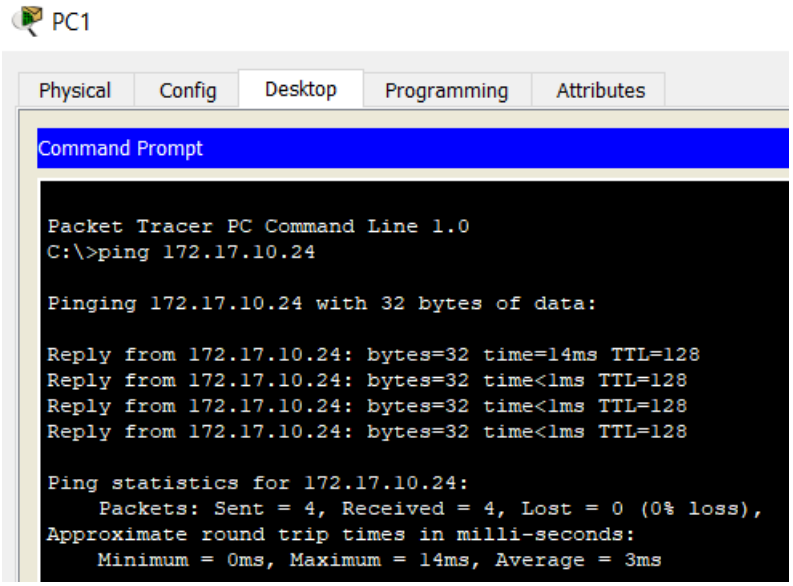


Фиг. 10.14. *Проверка на конфигурираните информационни канали за връзка при мрежово устройство S1*

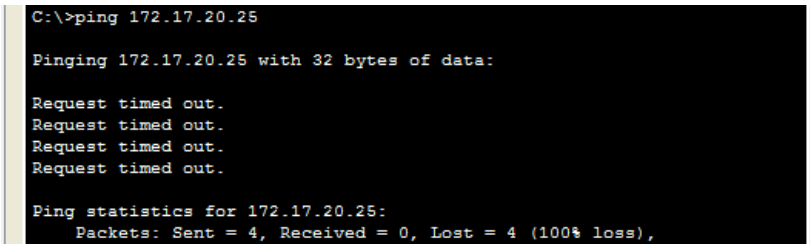
❖ Стъпка №11: Осъществяване на проверка чрез Ping процедури във връзка с коректността на комуникация и трансфера на инфомационните пакети от базовия суич S1 към S2 и S3 - показано на фиг. 10.15;



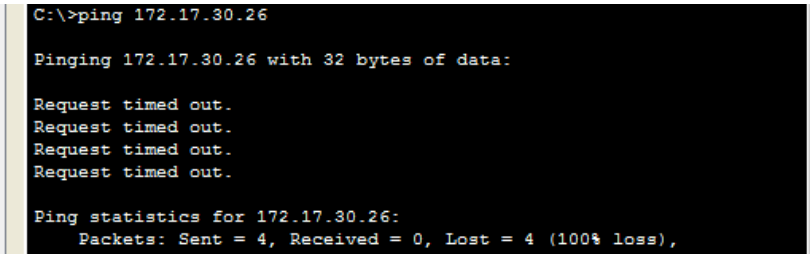
Фиг. 10.15. *Проверка на комуникацията в направление от S1 към S2 и S3*



а)



б)



в)

Фиг. 10.16. *Проверка на пакетното предаване на данни от PC1 към а), PC4 б) PC5 и в) PC6*

- Стъпка №12: Извършване на проверка на коректността на предаване на пакетни данни в режим „реално време“ между персоналните станции и обособените виртуални локални мрежи. В случая на фиг. 10.16 са реализирани дейности по проверка на статуса на информационния трансфер от:
  - PC1 към PC4 в рамките на VLAN 10 като комуникацията трябва да бъде успешна;
  - PC1 от VLAN 10 към PC5, който принадлежи към VLAN 20. Тук комуникацията следва да бъде безуспешна;
  - PC1 в структурата на VLAN 10 към PC6 в обхвата на VLAN 30 като отново трябва да бъде налице неуспешно получаване на пакетни данни.

  Допълнителните тестови процедури от PC2 или PC3 към персоналните станции PC4, PC5 и PC6 следва да бъдат направени по същия начин.

10.4. Контролни въпроси.

1. Какви видове Виртуални локални мрежи се прилагат при мрежовото изграждане на комуникационните и компютърните системи?
2. Кои са изискванията, определящи приложението на Гласовите VLANs?
3. Какви процедури се необходими при реализацията на Статичен, Динамичен и Гласов режим относно назначаването на switch портове към VLANs?
4. Дефинирайте предимствата на VLAN мрежовите инфраструктури, определящи тяхната приложимост в комуникациите и информационните системи.
5. Какъв процес се извършва с изпълнение на командата “swichport trunk native vlan”?

Лабораторно упражнение № 11

АДМИНИСТРИРАНЕ НА СИГУРНОСТТА ПРИ ПАКЕТНО ПРЕДАВАНЕ НА ДАННИ В СИМУЛАЦИОННО МОДЕЛИРАНИ СТРУКТУРИ ЧРЕЗ VPN ПОДХОДИ В СРЕДА CISCO PACKET TRACER

11.1. Цел на упражнението

Да се запознаят студентите с подходите за създаване и конфигуриране на VPN инфраструктури, както и различни категории от методи и алгоритми за криптиране и гарантиране на поверителността и оторизацията на потребителския достъп до предаваните пакетни данни.

11.2. Теоретична постановка

Посредством Виртуалните частни мрежи или VPN (Virtual Private Networks) се изгражда тунел, използвайки обществената Интернет мрежа, за отдалечена свързаност, комуникация и обмен на конфиденциални данни между мрежови потребители. Осъществява се емулиране на връзка от тип „точка до точка“ като създаденият „комуникационен тунел“ не зависи от реалното местоположение на персоналните потребителски компютърни станции.

IPsec тунелиране

IPsec или *IP Security* е една от най-често прилаганите технологии за изграждане на VPN и това е само една от многото причини всеки мрежови специалист да се запознае с нея. IPsec може да се разглежда като набор от протоколи (RFC 2401-2412), които не са обвързани с *определена технология за криптиране и хеширане и които позволяват да се изградят тунели с криптиран трафик (или само с автентификация и проверка на интегритета)*. Към момента това е една от най-често използваните VPN технологии. Структурните компоненти на IPsec са, както следва [21]:

- Протокол – възможностите са да се използва Encapsulation Security Payload (ESP) или Authentication Header (AH);
- Алгоритъм за осигуряване на конфиденциалност – най-често поддържаните криптографски алгоритми са DES, 3DES, AES и SEAL;
- Проверка на интегритета – избор на хеширащ алгоритъм, като най-често е възможно да се посочи MD5 или SHA;
- Генериране на ключовете – използване на цифрови сертификати или RSA;
- DH алгоритъм.

Компонентът AH предоставя възможности за:

- Автентификация на страните;
- Интегритет на данните,

докато ESP се използва за:

- Криптиране на потока от информация;
- Автентификация на страните;
- Интегритет на данните.

AH е IP протокол с номер 51 и за разлика от ESP (IP протокол 50) не може да шифрира данните. Ако IPsec VPN използва единствено AH, нивото на сигурност на комуникацията като цяло е ниско. При ESP първоначално се криптира пренасяния пакет, след което полученият резултат се използва като входни данни за HMAC алгоритъма. AH и ESP поддържат специална функционалност, предпазваща данните в тунела от тяхното повторно изпращане от трета страна – т.нар. "anti-replay" техника [21].

Конфиденциалността на данните се гарантира от протокола ESP и съответния криптографски алгоритъм за шифриране. Повечето IPsec системи предоставят възможност да се избере един от следните методи:

- *DES* – симетричен алгоритъм, използващ 56 битови ключове, който е с ниска криптографска надеждност и не се препоръчва да се използва в системи, които имат стриктни изисквания към сигурността;
- *3DES* – алгоритъм, базиран на DES, но използващ три последователни операции с DES и три различни ключа. В сравнение с DES има значително по-висока степен на сигурност;
- *AES* – симетричен алгоритъм по-сигурен от DES и по-бърз от 3DES;
- *SEAL* – поточен симетричен криптографски алгоритъм.

IPsec използва HMAC (Hashed Message Authentication Code), за да се гарантира интегритетът на пренасяните данни. Най-често поддържаните HMAC алгоритми са:

- *HMAC-MD5* – използва 128-битов ключ и хеширане чрез MD5, като резултатът е 128 битова стойност;
- *HMAC-SHA256* – на базата на 256-битов ключ и SHA хеширане се изчисляват 160 бита хеш резултат [22].

От криптографска гледна точка в сравнение с MD5, SHA е с по-висока степен на надеждност.

Един от най-важните етапи при изграждането на IPsec тунел е автентификацията на комуникиращите системи. По този начин се гарантира, че изграденият надежден тунел ще бъде между две легитимни устройства и данните няма да бъдат пренасочвани към трети лица. Двата варианта за автентификация при IPsec са:

- *Споделени ключове (PSK)* – двете системи имат предварително конфигурирани от администраторите "пароли" за изграждане на тунела. Споделената стойност се комбинира с други данни с цел еднозначна автентификация на отдалечената страна. Този метод е сравнително лесен за

конфигуриране, но поставя изисквания към начина на определянето и споделянето на PSK, както и води до липсата на гъвкавост при голям брой тунели;

- *RSA* – използването на асиметрични криптографски алгоритми и цифрови сертификати значително подобрява сигурността и точността на автентификацията, но е по-трудно за конфигуриране и изисква допълнителни проверки (CA) [22].

Приложението на симетричните алгоритми DES, 3DES, AES, както и на HMAC при IPsec изисква споделен ключ. Вместо тази стойност да бъде конфигурирана от администраторите (някои системи позволяват това) се използва DH протоколът за надеждно генериране на криптографски ключове през несигурен канал за обмяна на пакети. DH групите при IPsec най-често са:

- *DH групи 1, 2 и 5* – използват се за създаване на ключове с размери 768, 1024 и 1536 бита, като не се препоръчва те да бъдат използвани;

- *DH групи 14, 15 и 16* – генерираните ключове са с дължини от 2048, 3072 и 4096 бита;

- *DH групи 19, 20 и 24* – използват ECC и поради спецификата на тези алгоритми се генерират надеждни ключове с малка дължина, респективно 256, 384 и 2048 бита. Препоръчва се използването на DH24, ако системите позволяват.

Някои от криптографските алгоритми изискват да има налични споделени ключове, които за да се генерират надеждно през несигурния комуникационен канал се прилага алгоритъмът DH. При IPsec се използва протоколът “Internet Key Exchange” (IKE), при който на базата на няколко обменени пакета между двете системи изграждащи тунела се изчисляват споделени ключове. IKE е описан в RFC 2409 и използва транспортния протокол UDP и порт 500. Най-общо за IKE може да се каже, че това е хибриден протокол, който обхваща “Internet Security Association and Key Management Protocol” (ISAKMP) и механизма за обмяна на ключове “Oakley and Skeme”. IKE работи в две фази:

1. *IKE фаза 1* – първоначално определяне на основните настройки за подсигуряване на тунела, включващи автентификация на системите, криптографски алгоритми, генериране на ключове и др.;

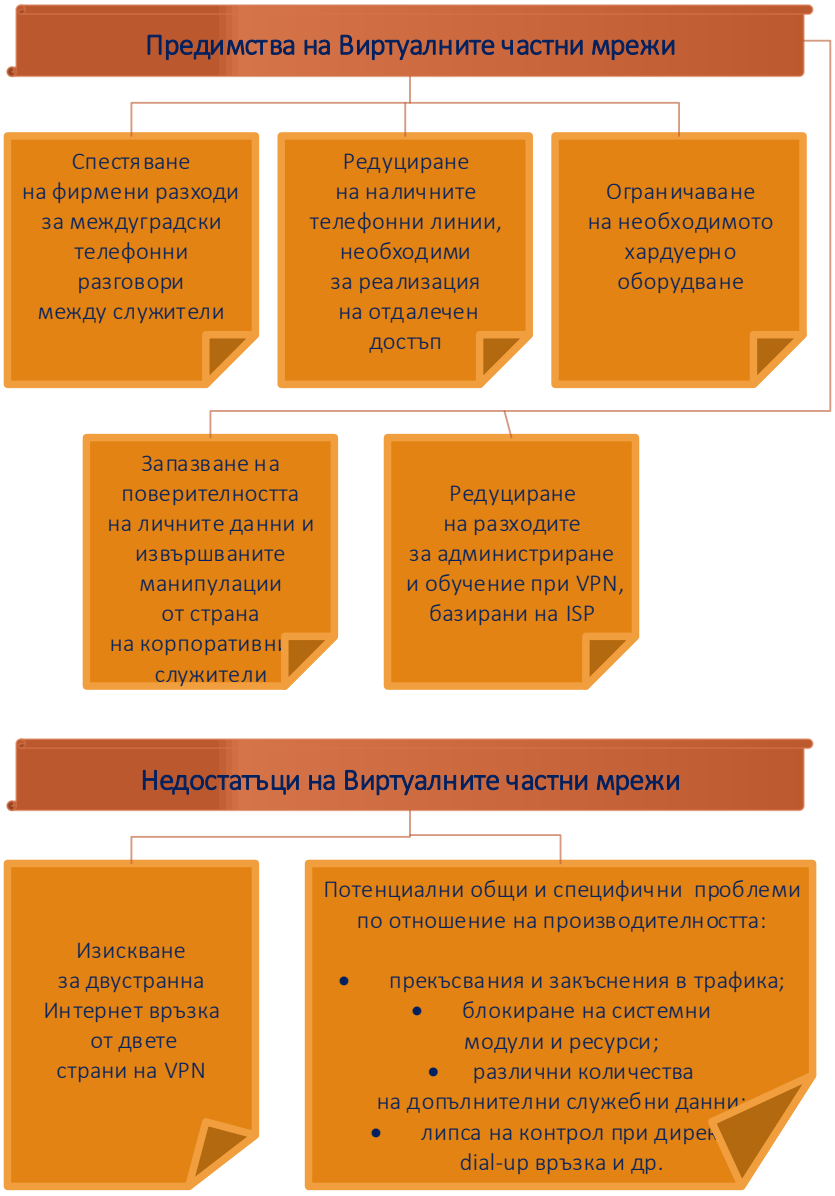
2. *IKE фаза 2* – след успешно преминаване на фаза 1 при необходимост от предоговаряне на параметрите се стартира значително по-бързата фаза 2.

Изграждане на VPN системи, преимущества и недостатъци

Изграждането на Виртуални частни мрежи има отношение към постигане на централизирано управление на мрежовите и онлайн ресурси, защита на оперативните информационни масиви и управление на достъпа за оторизираните корпоративни потребители.

VPNs осигуряват възможности за осъществяване на отдалечен достъп до фирмени служители, клиентски бизнес организации или връзка между фирмени офиси, намиращи

се на различни географски местоположения. Концептуалните предимства и недостатъци от интеграция на VPN са представени на фиг. 11.1 [21, 22].



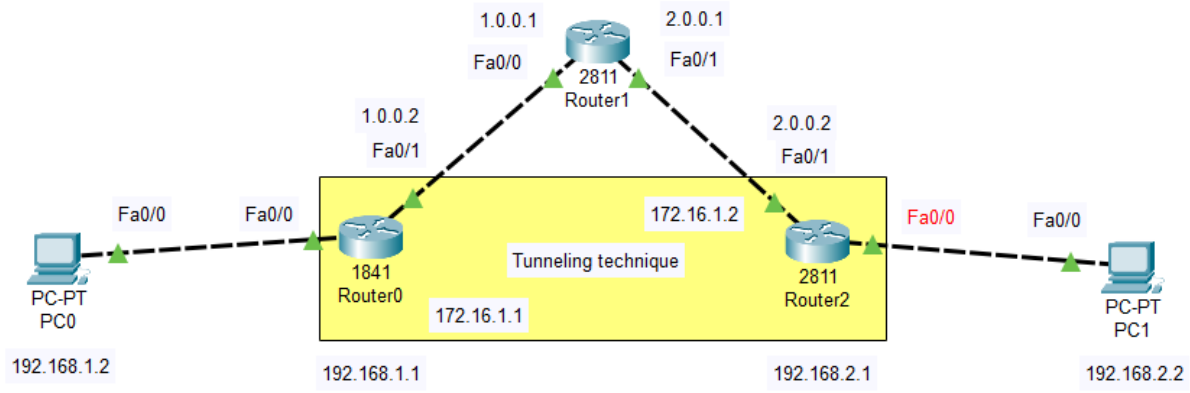
Фиг. 11.1. *VPN предимства и недостатъци*

11.3. Последователност на работа и задачи за изпълнение

11.3.1. Конфигуриране на Виртуална частна мрежа със статична пакетна маршритизация в Cisco Packet Tracer

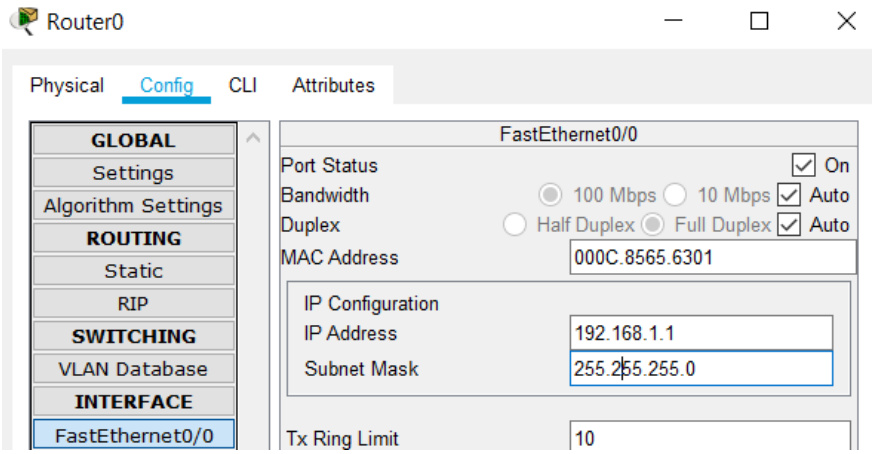
Една от възможностите за изграждане на виртуални чатни мрежи се свързва с конфигуриране на статичното предаване на информационни пакетни единици. За целта може да бъде приложена следната стъпкова последователност, съответно:

- Изграждане на мрежовата VPN топология на основата на маршрутизатори 1841 и прилагане на обратно свързване между мрежовите устройства, представено на фиг. 11.2;

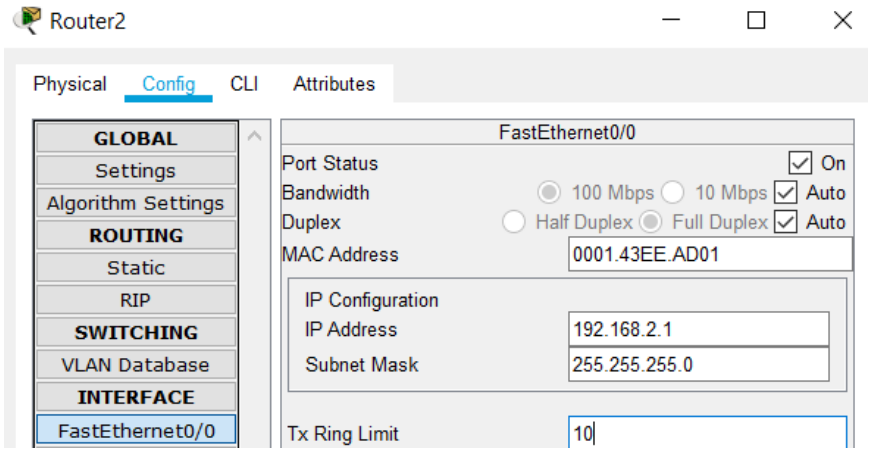


Фиг. 11.2. *VPN мрежова конфигурация с прилагане на tunneling подход*

- IP конфигурация на маршрутизатори Router0 и Router2 относно FastEthernet интерфейси Fa0/0, както е показано на фиг. 11.3;



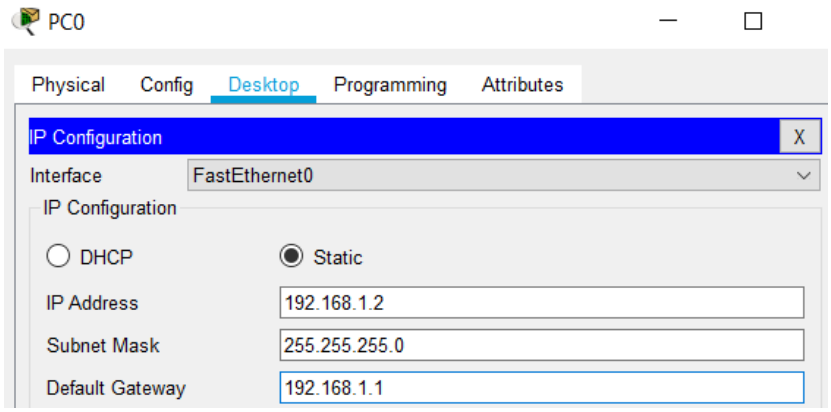
а)



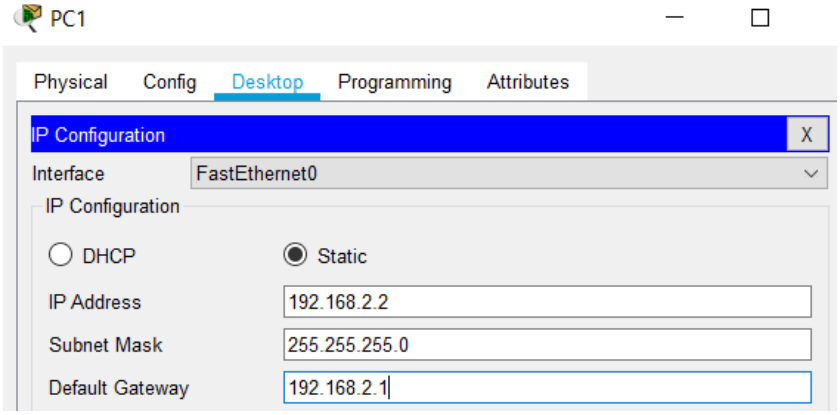
б)

Фиг. 11.3. *IP настройки на fa0/0 при а) Router0 и б) Router2*

- Присвояване на IP адрес, подмрежова маска и Gateway при персонални поребителски компютърни станции PC0 и PC1, проведено и илюстрирано на фиг. 11.4;



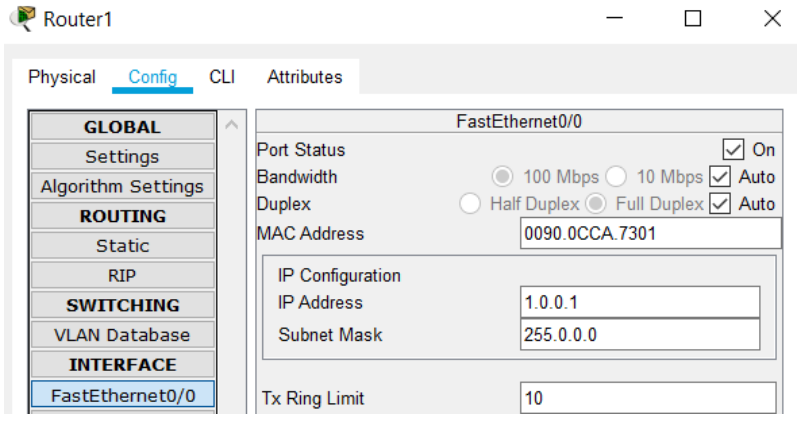
а)



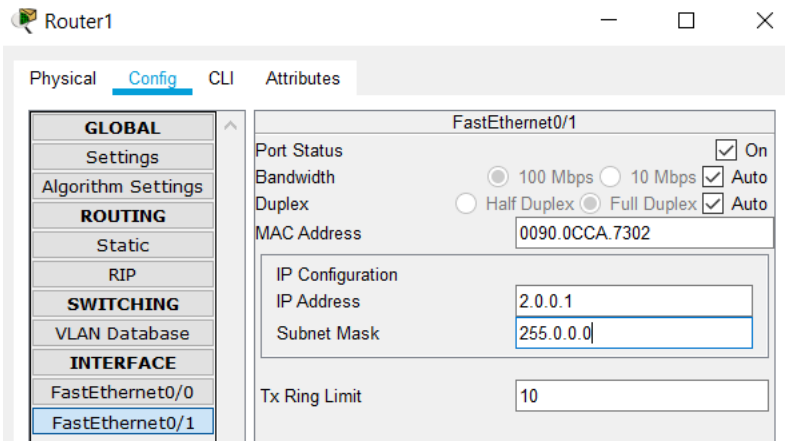
б)

Фиг. 11.4. *IP конфигурация на а) PC0 и б) PC1*

- IP сегменто мрежово конфигуриране на Fast Ethernet комуникационни интерфейси Fa0/0 и Fa0/1 във връзка с маршрутизиращо звено Router1, представено на фиг. 11.5;



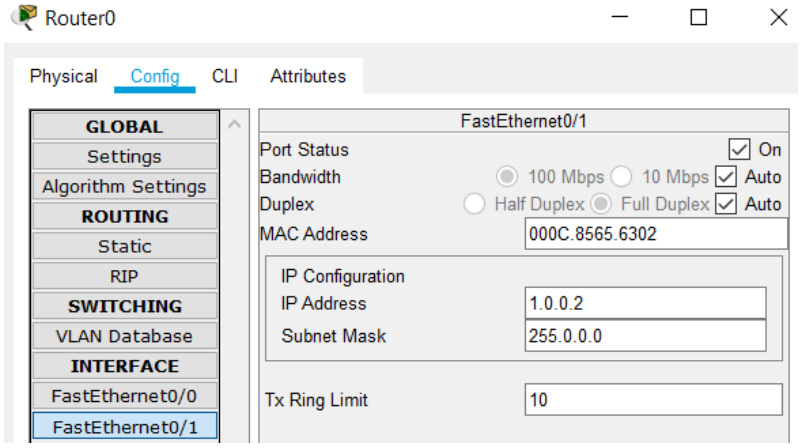
а)



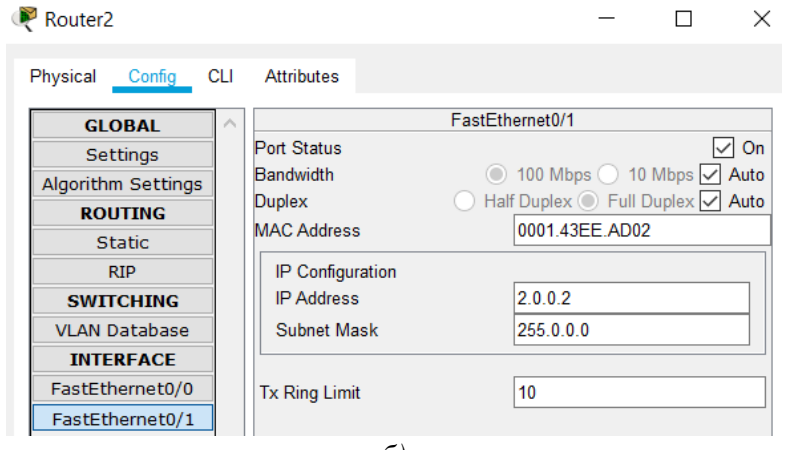
б)

Фиг. 11.5. *IP настройки при Router1 относно а) fa0/0 и б) fa0/1*

- Фиксиране на мрежови IP адрес и подмрежова маска относно интерфейси Fast Ethernet Fa0/1 при маршрутизарори Router0 и Router2, реализирано на фиг. 11.6;



а)

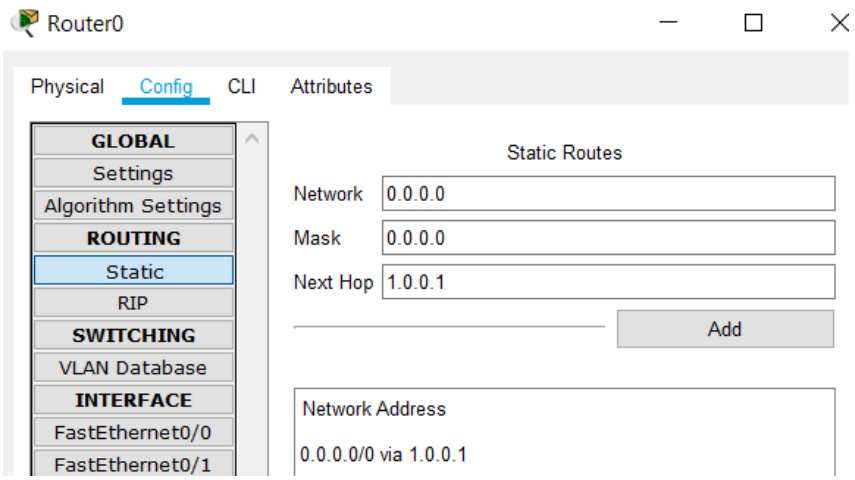


б)

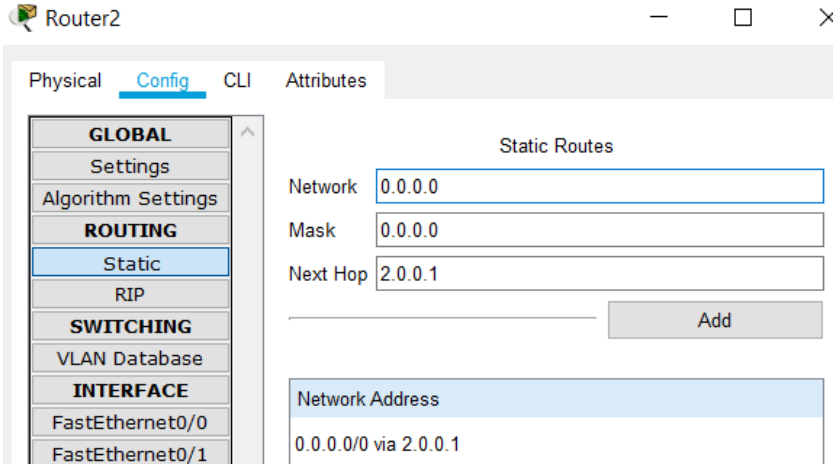
Фиг. 11.6. *Настройка на fa0/0 при маршрутизатори а) Router0 и б) Router2*

- Назначаване на информационни направления за статичен тип пакетна маршрутизация при мрежови звена Router0 и Router2, както следва “Network”, “Submask” и “Next flop”. Конфигурационният параметър се отнася до „следващ

мрежови скок“ или адрес на следващото устройство, към което ще бъдат изпратени пакетни данни. Стъпката е онагледена на фиг. 11.7;



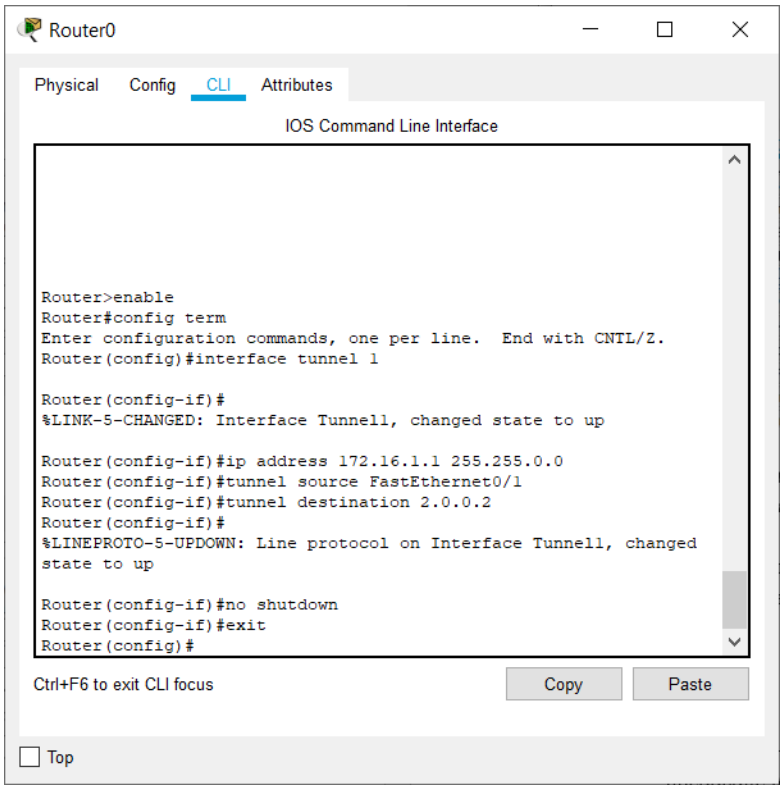
а)



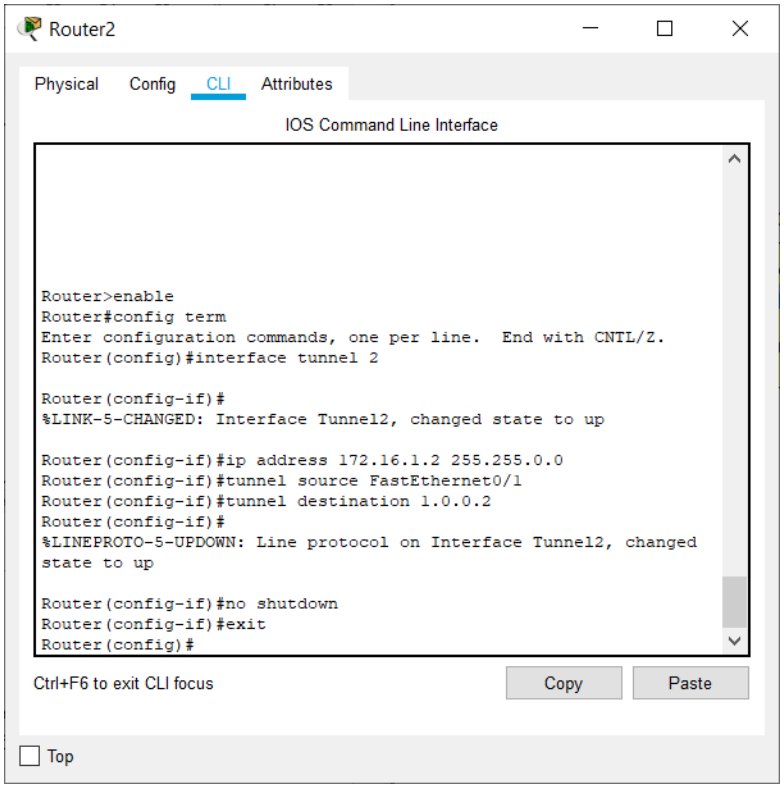
б)

Фиг. 11.7. *Конфигуриране на направления за статичен тип маршрутизация относно а) Router0 и б) Router2*

- Дефиниране на параметрите на процеса на „*Мрежово тунелиране*“ в средата на CLI интерфейса, касаещо маршрутизатори Router0 и Router2. Тук се указват информационният източник и крайната дестинация на предаване на капсулираните данни. Дейностите са показани на фиг. 11.8;



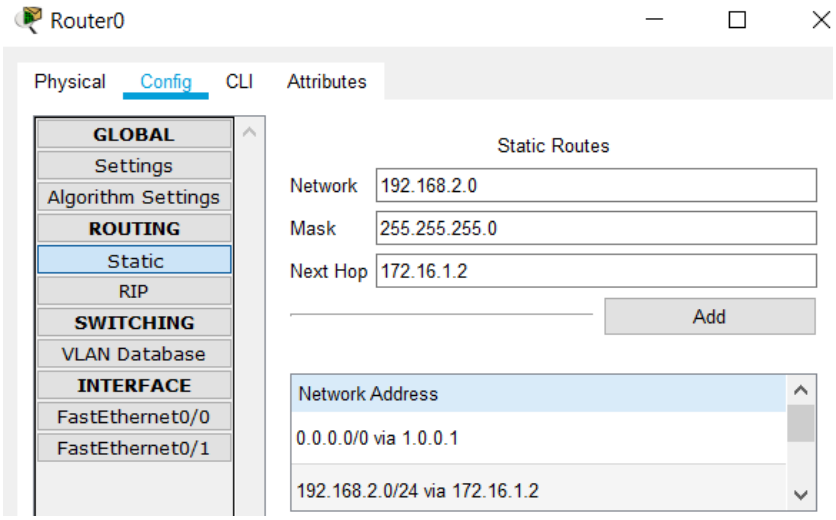
а)



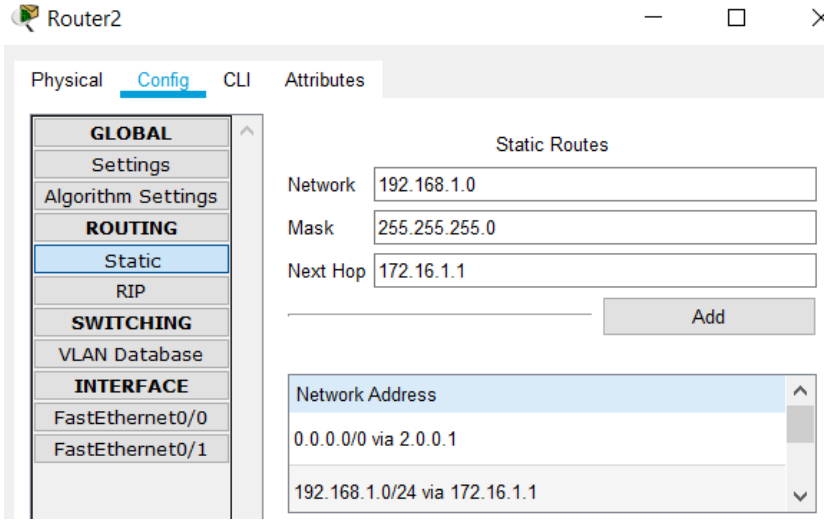
б)

Фиг. 11.8. *Денифиране на процеса „Мрежово тунелиране“ относно пакетното предаване на информационни единици за а) Router0 и б) Router2*

- Включване на допълнителни информационни направления за целите на статичната маршрутизация за трансфер на капсулирани данни по установения „информационен тунел“ във връзка с мрежовите устройства Router0 и Router2, показано на фиг. 11.9;



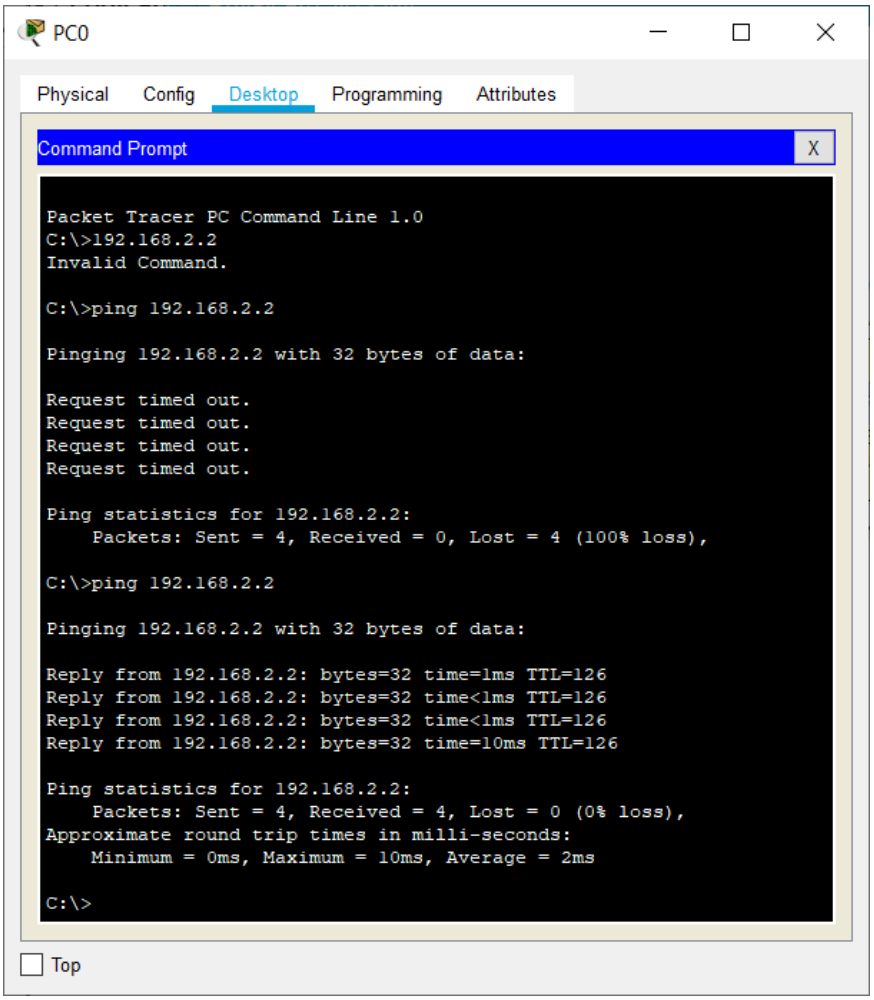
а)



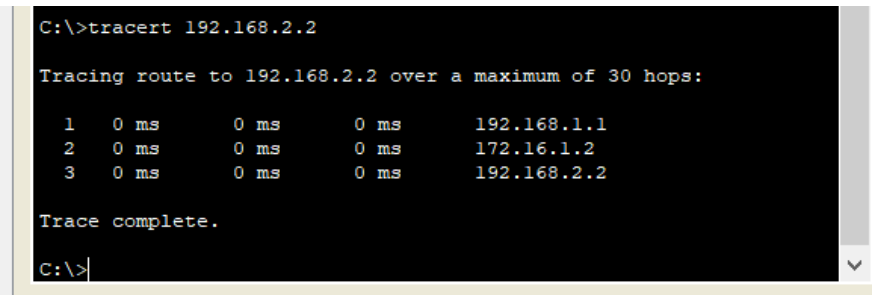
б)

Фиг. 11.9. *Назначаване на допълнителни информационни направления за статична маршрутизация при а) Router0 и б) Router2*

- Тестване на коректността и проследяване на пакетното предаване на данни между персонални компютърни станции PC0 и PC1 с помощта на приложение Command Prompt (забележка: Необходимост от повторение от процедурата в случай на първоначално регистрирано неуспешно предаване!). Процедурата е представена на фиг. 11.10;

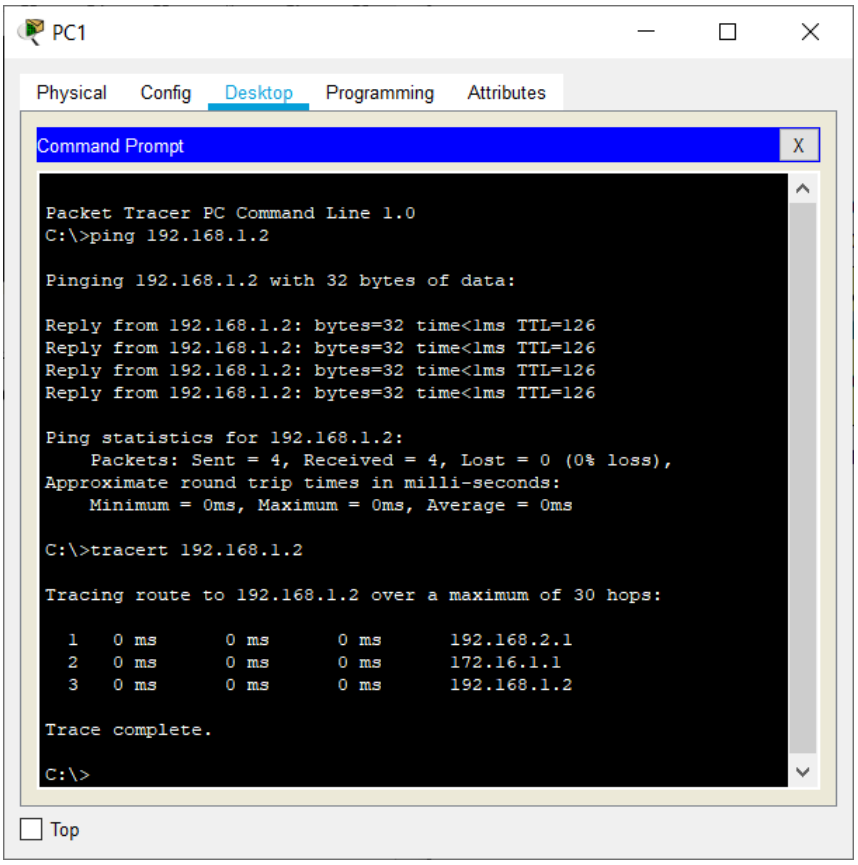


а)



б)

Фиг. 11.10. *Тестване на коректността а) и проследяване на пакетното предаване на данни б) между PC0 и PC1*



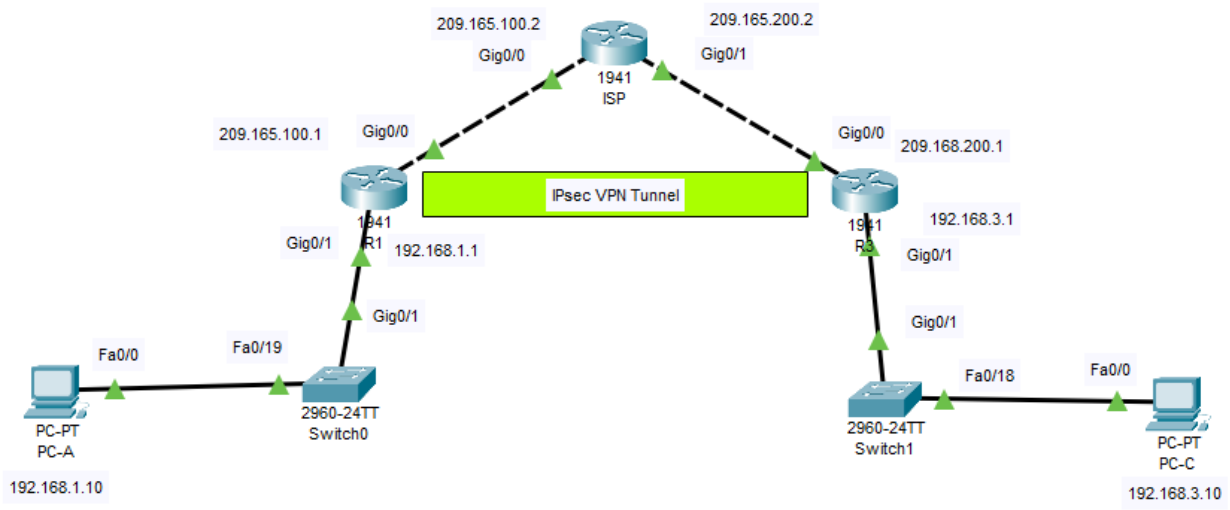
Фиг. 11.11. *Тестване на коректността и проследяване на пакетното предаване на данни между PC1 и PC0*

- Проверка на състоянието по отношение на пакетното предаване на данни между персонални компютърни станции PC1 и PC0, както и проследяване на преходите между мрежовите устройства по комуникационното трасе, илюстрирана на фиг. 11.11.

11.3.2. Конфигурация на VPN с IPsec тунериране в програмната среда Cisco Packet Tracer

IPsec тунелирането може да бъде реализирано в следната стъпкова последователност:

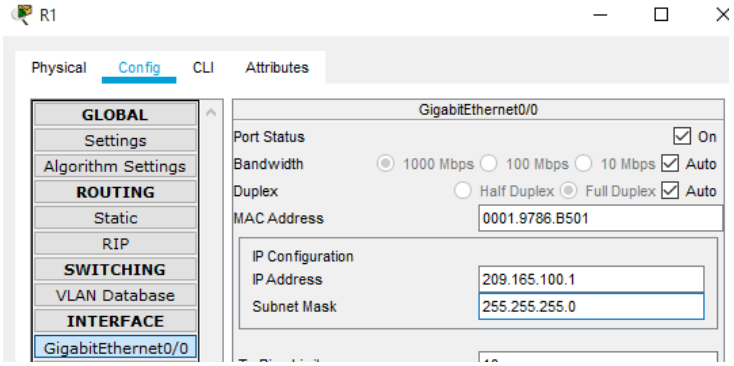
- Изграждане на и свързване на устройствата от обектната мрежова VPN архитектура с последващо IPsec тунелиране, показана на фиг. 11.12;



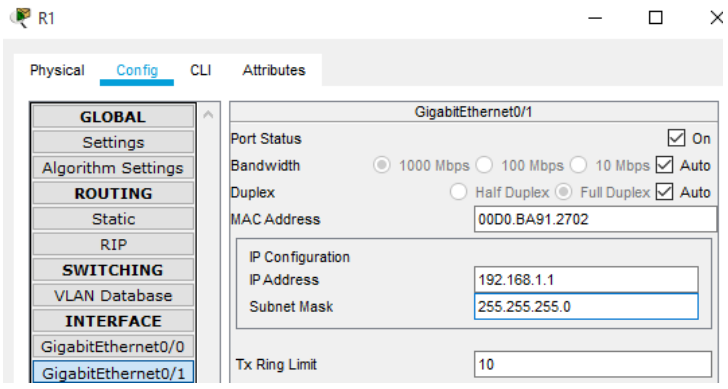
Фиг. 11.12. *VPN IPsec мрежова конфигурация*

- Настройка на комуникационните Gigabit Ethernet интерфейси по отношение на IP адрес и подмрежова маска, както следва при:

  ✓ Маршрутизатор R1 (фиг. 11.13):



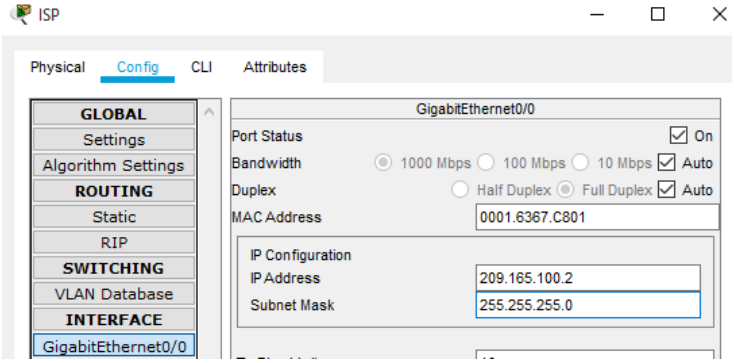
а)



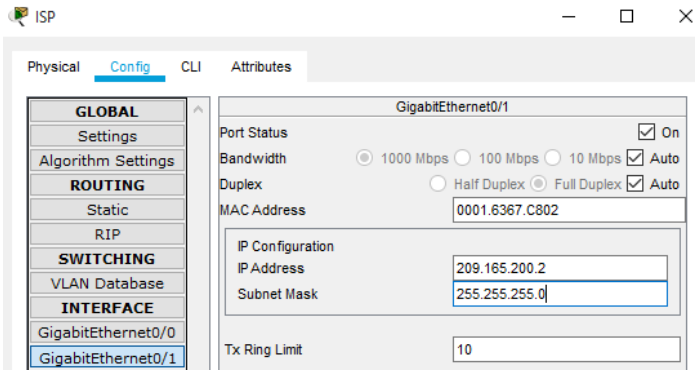
б)

Фиг. 11.13. *Настройка на интерфейси а) GigabitEthernet0/0 и б) GigabitEthernet0/1 при R1*

✓ Маршрутизатор ISP, както е направено на фиг. 11.14:



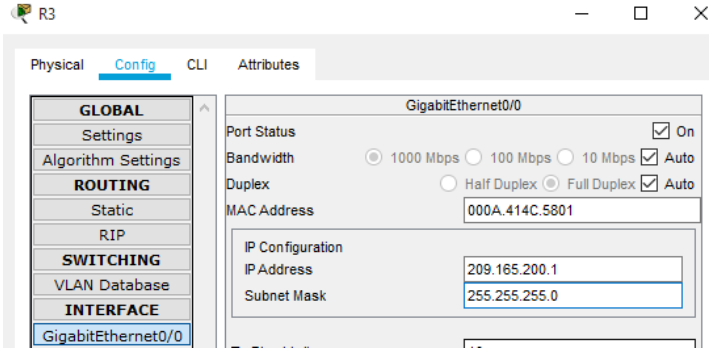
а)



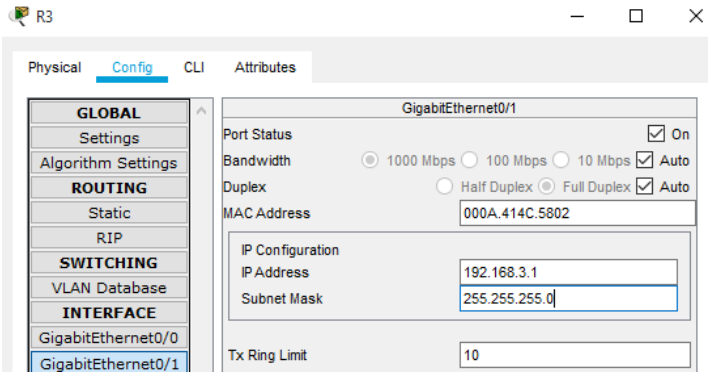
б)

Фиг. 11.14. *Настройка на интерфейси а) GigabitEthernet0/0 и б) GigabitEthernet0/1 при ISP*

✓ Маршрутизатор R3, както е показано на фиг. 11.15:



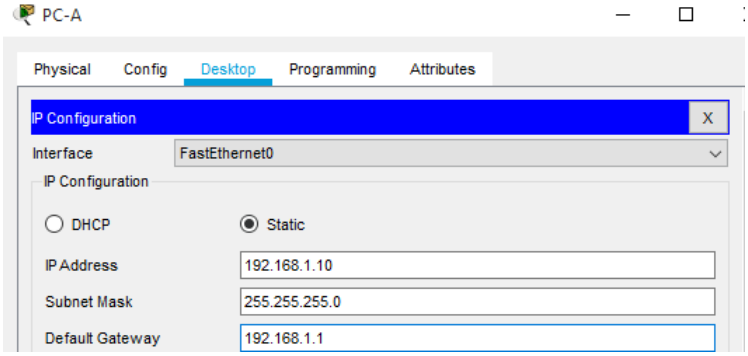
а)



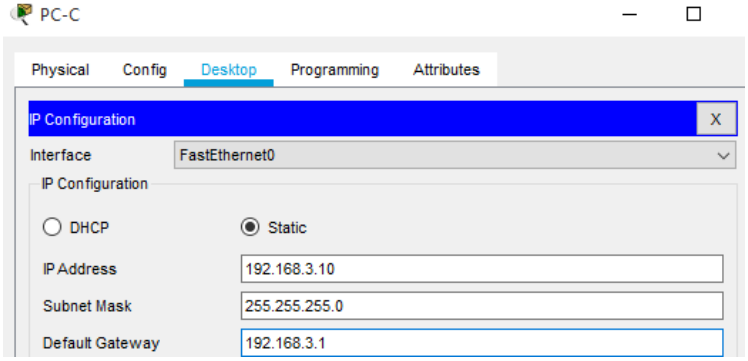
б)

Фиг. 11.15. *Настройка на интерфейси а) GigabitEthernet0/0 и б) GigabitEthernet0/1 при R3*

- Изграждане на логическа връзка или IP конфигурация във връзка с потребителските персонални компютърни станции PC-A и PC-C, показано на фиг. 11.16;



а)



б)

Фиг. 11.16. *Конфигурация на а) PC-A и б) PC-C*

- Активиране на security-k9 модул относно изграждане на IPsec VPN Tunnel при маршрутизатори R1 и R3, реализирано на фиг. 11.17. Процедурата се изразява в разрешаване на Security Technology Package licence като обхваща:
  - ✓ „последващо зареждане на маршрутизатора“;
  - ✓ „приемане на лиценза“;
  - ✓ „съхраняване на конфигурацията“;
  - ✓ „конфигурационно презареждане“.

```
R1>enable
R1#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#license boot module c1900 technology-package securityk9
```

а)

```
ACCEPT? [yes/no]: yes
% use 'write' command to make license boot config take effect on next
boot

R1(config)#: %IOS_LICENSE_IMAGE_APPLICATION-6-LICENSE_LEVEL: Module
name = C1900 Next reboot level = securityk9 and License = securityk9

R1(config)#exit
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
```

б)

```
R1#reload
Proceed with reload? [confirm]
System Bootstrap, Version 15.1(4)M4, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 2010 by cisco Systems, Inc.
Total memory size = 512 MB - On-board = 512 MB, DIMM0 = 0 MB
CISCO1941/K9 platform with 524288 Kbytes of main memory
Main memory is configured to 64/-1(On-board/DIMM0) bit mode with ECC
disabled

Readonly ROMMON initialized

program load complete, entry point: 0x80803000, size: 0x1b340
program load complete, entry point: 0x80803000, size: 0x1b340

IOS Image Load Test
___________________
Digitally Signed Release Software
program load complete, entry point: 0x81000000, size: 0x2bb1c58
Self decompressing the image :
########################################
```

в)

Фиг. 11.17. *Security-k9 license – а) активиране, б) съхранение и в) презареждане*

- Назначаване на интерфейсни направления за пакетно предаване на данни между мрежовите маршрутизатори. Стъпката, онагледена на фиг. 11.18, се извършва последователно при маршрутизиращи звена R1 и R3;

```
R1>enable
R1#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip route 0.0.0.0 0.0.0.0 209.165.100.2
R1(config)#exit
```

а)

```
R3>enable
R3#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#ip route 0.0.0.0 0.0.0.0 209.165.200.2
R3(config)#exit
```

б)

Фиг. 11.18. *Направления на пакетно предаване на данни при а) R1 и б) R3*

- Прилагане на ACL (Access List Configuration) процедура по управление на достъпа и назначаване на направления на информационния трафик по отношение на мрежите, касаещи маршрутизатори R1 и R3 - показана на фиг. 11.19;

```
R1>enable
R1#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#access-list 100 permit ip 192.168.1.0 0.0.0.255 192.168.3.0 0.0.0.255
```

а)

```
R3>enable
R3#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#access-list 100 permit ip 192.168.3.0 0.0.0.255 192.168.1.0 0.0.0.255
```

б)

Фиг. 11.19. *Разрешаване на достъпа при а) R1 и б) R3*

- ISAKMP ФАЗА „1“ във връзка с подсигуряване на тунела за защита на ISAKMP пакетни съобщения: Дефиниране на ISAKMP политика за сигурност относно

мрежовите сегменти в зоната на PC-A и PC-C, респективно при маршрутизатори R1 и R3. Процедурата, онагледена на фиг. 11.20, включва:

- ✓ дефиниране на симетричен криптиращ алгоритъм AES с 256 битов ключ;
- ✓ прилагане на споделен ключ за автентикация;
- ✓ генериране на DH група 5 криптографски ключ с дължина 1536 бита.

```
R1(config)#crypto isakmp policy 10
R1(config-isakmp)#encryption aes 256
R1(config-isakmp)#authentication pre-share
R1(config-isakmp)#group 5
R1(config-isakmp)#
```

а)

```
R3#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#access-list 100 permit ip 192.168.3.0 0.0.0.255 192.168.1.0 0.0.0.255
R3(config)#crypto isakmp policy 10
R3(config-isakmp)#encryption aes 256
R3(config-isakmp)#authentication pre-share
R3(config-isakmp)#group 5
```

б)

Фиг. 11.20. *Конфигуриране на ISAKMP policy при а) R1 и б) R3*

- ISAKMP ФАЗА „1“: Назначаване на ISAKMP key, за конкретния случай “secretkey”, във връзка със съответен адрес за маршрутизация. Процедурата се изпълнява при R1 и R3, както е показано на фиг. 11.21;

```
R1>enable
R1#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#crypto isakmp key secretkey address 209.165.200.1
```

а)

```
R3(config-isakmp)#group 5
R3(config-isakmp)#exit
R3(config)#crypto isakmp key secretkey address 209.165.100.1
R3(config)#
```

б)

Фиг. 11.21. *Задаване на ISAKMP key при а) R1 и б) R3*

- IPsec ФАЗА „2“ се извършва създаване на Transform set – комбинация от индивидуални IPsec трансформации за назначаване на специфицирани политики за сигурност във връзка със защита на трафика (създаване на тунел за защита на трафичните данни), използвайки:
  - ✓ esp-aes – ESP протокол за конфиденциалност на данни и AES симетричен алгоритъм;

- ✓ esp-sha-hmac – ESP протокол за конфиденциалност на данните и подход на базата на SHA хеширане и 256 ключ по отношение на интегритета на данните.

Указаната стъпка се прилага по направления от R1 към R3 и от R3 към R1, както е онагледено на фиг. 11.22;

```
R1#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#crypto ipsec transform-set R1->R3 esp-aes 256 esp-sha-hmac
R1(config)#
```

а)

```
R3#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#crypto ipsec transform-set R3->R1 esp-aes 256 esp-sha-hmac
R3(config)#
```

б)

Фиг. 11.22. *Промяна на подхода на криптиране - а) R1 към R3 и б) R3 към R1*

- Създаване на специализирана Crypto map “IPSEC-MAP” с унифициран номер „10“ и процедурна идентификация като типова ipsec-isakmp map. Във връзка с процедурата, дадена на фиг. 11.23, се извършва следната последователност от процеси по:

  - ✓ назначаване на връзка към съответен маршрутизатор R1 (R3);
  - ✓ използване на PFS (Perfect Forward Secrecy) инструмент, касаещо генериране на нов секретен DH ключ (DH група 5 криптографски ключ с дължина 1536 бита) за криптиране на IPsec трафика;
  - ✓ фиксиране на времевата продължителност на прилагане на създадените ключове за криптиране на трафичните данни в създадения IPsec тунел (IPSec Security Associations – IPsec SAs);
  - ✓ прилагане на реглементираните IPsec трансформации;
  - ✓ назначаване на регламентирания трафик посредством Access List Configuration.

```
R1#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#crypto map IPSEC-MAP 10 ipsec-isakmp
% NOTE: This new crypto map will remain disabled until a peer
        and a valid access list have been configured.
R1(config-crypto-map)#set peer 209.165.200.1
R1(config-crypto-map)#set pfs group5
R1(config-crypto-map)#set security-association lifetime seconds 86400
R1(config-crypto-map)#set transform-set R1->R3
R1(config-crypto-map)#match address 100
R1(config-crypto-map)#
```

а)

```
R3>enable
R3#config term
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#crypto map IPSEC-MAP 10 ipsec-isakmp
% NOTE: This new crypto map will remain disabled until a peer
        and a valid access list have been configured.
R3(config-crypto-map)#set peer 209.165.100.1
R3(config-crypto-map)#set pfs group5
R3(config-crypto-map)#set security-association lifetime seconds 86400
R3(config-crypto-map)#set transform-set R3->R1
R3(config-crypto-map)#match address 100
R3(config-crypto-map)#
```

б)

Фиг. 11.23. *Създаване на Crypto map IPSEC-MAP при маршрутизатори а) R1 и б) R3*

- Прилагане на създадената Crypto map със зададено наименование IPSEC-MAP – процедурата се изпълнява във връзка с комуникационните интерфейси GigabitEthernet0/0. Съдържанието на процедурата е представено на фиг. 11.24;

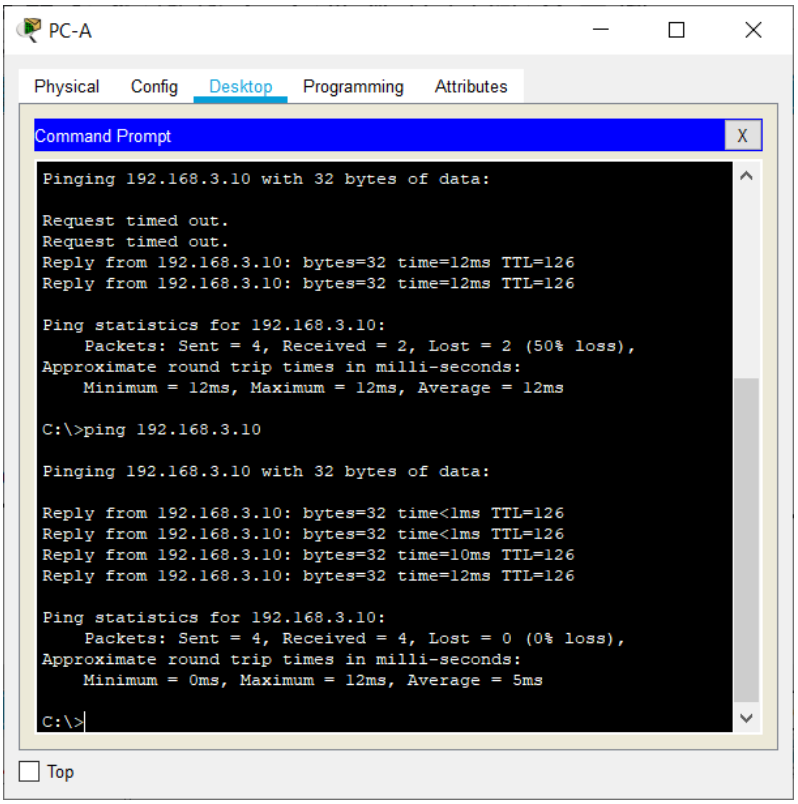
```
R1(config)#int g0/0
R1(config-if)#crypto map IPSEC-MAP
*Jan  3 07:16:26.785: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
R1(config-if)#
```

а)

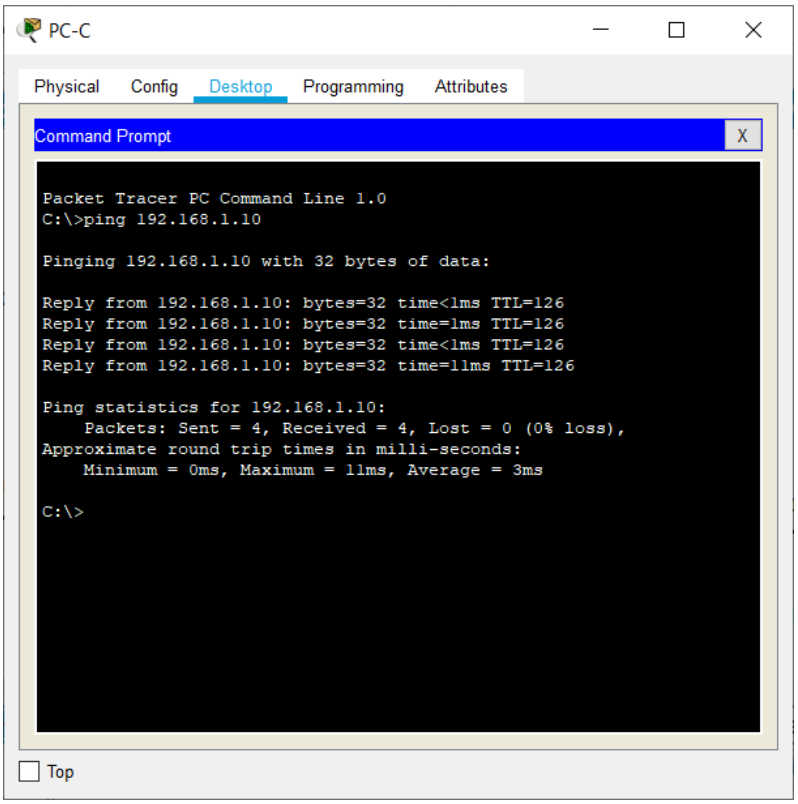
```
R3(config)#int g0/0
R3(config-if)#crypto map IPSEC-MAP
*Jan  3 07:16:26.785: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
R3(config-if)#
```

б)

Фиг. 11.24. *Прилагане на създадената Crypto map IPSEC-MAP относно комуникационния интерфейс g0/0 при маршрутизатори а) R1 и б) R3*



а)



б)

Фиг. 11.25. *Предаване на PDU инфомационни единици чрез изградения IPsec VPN тунел по направление а) „PC-A към PC-C“и б) „PC-C към PC-A“*

- Тестване на пакетното предаване на данни между персоналните компютърни станции в обхвата на изградения IPsec тунел чрез конкретни ping процедури по направления „PC-A към PC-C“ „PC-C към PC-A“, както е показано на фиг. 11.25.

11.4. Контролни въпроси

1. Кои са градивните структурни компоненти на подхода за тунелиране IPsec?
2. Какво е предназначението на компонета ESP при IPsec подход?
3. Кои методи за криптиране на информацията се използват при IPsec?
4. Кои са най-често прилаганите криптиращи HMAC алгоритми за гарантиране на „интергритета“ на данните?
5. Какви са основните DH категории при IPsec и с какви дължини на криптиращите ключове се свързват?
6. Опишете принципа на работа на IKE фазите при прилагането на IPsec?
7. Какви са предимствата и недостъците на VPN?

Лабораторно упражнение № 12

МОДЕЛИРАНЕ И КОНФИГУРИРАНЕ НА IOT СЕНЗОРНИ МРЕЖОВИ СТРУКТУРИ С ИМИТИРАНИ МНОГОФУНКЦИОНАЛНИ СЕНЗОРНИ МОДУЛИ В CISCO PACKET TRACER

12.1. Цел на упражнението

Да се запознаят студентите с възможностите на програмната среда Cisco Packet Tracer за моделиране и програмиране на сензорни мрежи на основата на многофункционални модули с интеграция на IoT технология.

12.2. Теоретична постановка

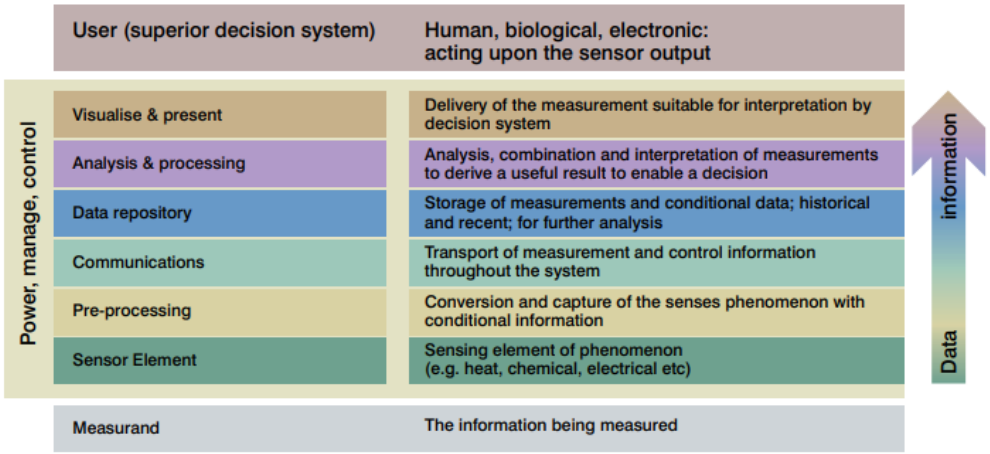
Интелигентни сензорни мрежи и системи

Сензорните мрежи имат множество приложения като изследванията и разработките на сензорната мрежа извличат много концепции и протоколи от разпределени компютърни мрежи като Интернет. Наред с това съществуват няколко технически предизвикателства. Изграждането им трябва да бъде съобразено спрямо специализираното естество на сензорите и фактора, че много сензорни мрежови приложения могат да включват отдалечени мобилни сензори с ограничени източници на енергия, които трябва динамично да се адаптират към средата си [23].

На фиг. 12.1 са дефинирани основните положения, свързани с изграждането на сензорни системи и направлението на информационните потоци в тях. Тук са включени процеси, като:

- „Измерване“;
- „Избор на сензорни елементи“;
- „Предварителна обработка на измервателните данни“;
- „Обезпечаване на комуникационния трансфер на данни“;
- „Осигуряване на информационни хранилища“;
- „Анализ и обработка на измервателната информация“;
- „Информационна визуализация и представяне“,

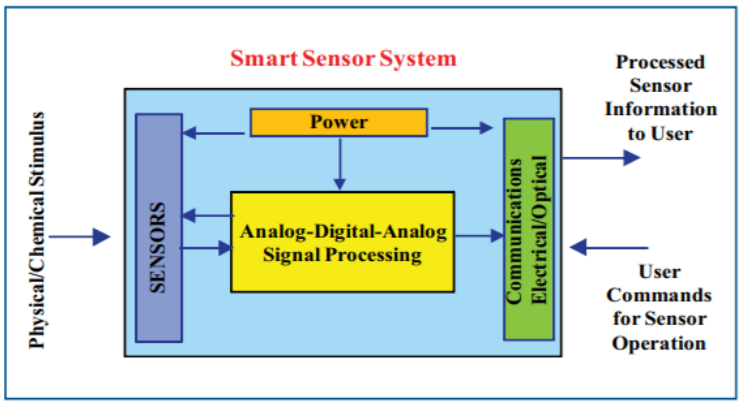
за които следва да бъде избрани хардуерно и софтуерно обезпечаване.



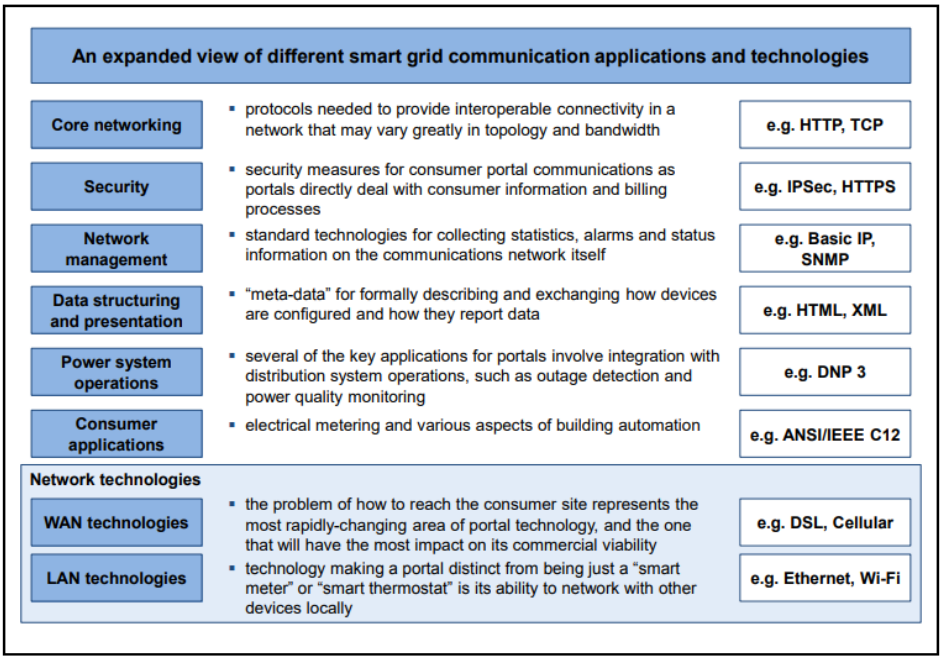
Фиг. 12.1. *Направление на данните и основни процеси прилагани върху тях при изграждане на сензорни системи*

Примерна архитектура на „smart sensor system“ е представена на фиг. 12.2. Системата се състои от следните компоненти, управлявани посредством специализиран микроконтролер:

- Сензорни елементи, реагиращи на „физически“ или „химически“ въздействия;
- Захранващ блок;
- ADC/DAC блокове и модул за сигнална обработка;
- Комуникационен блок за предаване на потребителски команди за сензорно управление и сензорна информация на базата Electrical/Optical преносна среда [23].



Фиг. 12.2. *Архитектура на smart sensor system*



Фиг. 12.3. *Класификация на интелигентните комуникационни приложения и технологии*

В тази връзка в таблица 12.1 са показани предимствата и недостатъците на основните WAN технологии при тяхно внедряване в интелигентните мрежи за пренос на сензорна информация. Информацията, предоставена от интелигентните сензори и интелигентните измервателни уреди, трябва да бъде предавана посредством сигурно комуникационно трасе. Комуникационната архитектура е необходимо да поддържа висикоскоростен и

двупосочен информационен трансфер. Различните комуникационни приложения и технологии определят изграждането на ефективни и надеждни комуникационни трасета. Според някои източници може да бъде направена класификацията на информационните услуги, показана на фиг. 12.3. Фигурата предоставя обща характеристика на използваните мрежови технологии с голям обхват (WAN) и локални приложения (LAN). WAN осигуряват средство за двупосочен информационен поток в интелигентната мрежа. Налични са множество технологии, които осигуряват широколентови и теснолентови решения за изграждане на интелигентни мрежи.

Таблица 12.1. Предимства и недостатъци на WAN технологиите

WAN technology		Strengths	Weaknesses
ADSL (Asymmetric Digital Subscriber Line)		• high availability • consistent bandwidth regardless of number of users and use in time	• decreasing bandwidth with distance
Cable modem		• high bandwidth • high availability	• inconsistent bandwidth depending on number of users and time of day
FTTH (Fiber to the Home)		• scalability • high bandwidth • planned security measures	• relatively high costs • no deployment in rural areas
WiMAX (IEEE 802.16)		• does not require deployment of a costly wired infrastructure	• early stage of deployment, uncertain whether the technology will meet its range targets
Power line communications	**BPL** (broadband over power line)	• existing wired infrastructure (particular advantage in rural areas)	• cost of deployment[3] • BPL not suited for particular applications as it is dependent on current on the power line • mostly proprietary
	Narrowband PLC (e.g. IEC 61 334-5 PLC)	• field-proven in Europe • international standards (mostly European)	• cost of deployment[4] • not suited for particular applications as it is dependent on current on the power line
Cellular Services		• high coverage area • potentially low costs	• fast development of new technology (danger of being tied to one provider) • some packet-switched services not very reliable • security concerns • some systems may not transmit unsolicited data
Satellite Services		• universally available, regardless of concrete location	• high costs • low effective bandwidth • additional security measures required • low reliability during bad weather conditions
Paging Systems		• ubiquity • low costs • reliability	• low bandwidth and thus only support of a few applications such as simple emergency alerts

LAN технологиите реализират свързаност между различни интелигентни устройства като могат да бъдат категоризирани в следните групи:

- wireless IEEE standards 802.x;
- wired Ethernet;
- in-building power line communications.

В таблица 12.2 е показана сравнителна характеристика между основните wi-fi IEEE стандарти за пренос на данни относно приложения, силни и слаби страни [23, 24].

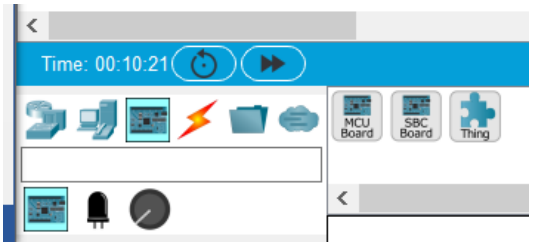
Таблица 12.2. Сравннение между основни wi-fi IEEE стандарти

IEEE standard	Applications	Strengths	Weaknesses
Wi-Fi (IEEE 802.11)	• connecting equipment at customer' site • access between WAN networks and customers' site	• easy deployment • falling costs	• only useful within the customer site • additional security layers required
ZigBee (IEEE 802.15.4)	• drive-by meter reading • user interface at customers' site • connection of sensors and other equipment in a customer LAN	• low power requirements • low implementation cost • good scalability (many devices can be connected) • particularly designed for use in industrial and home automation or security applications	• limited range • relatively low data rates (but probably sufficient) • possibly more secure than other standards
Bluetooth (IEEE 802.15.1)	• drive-by meter reading • user interface at customers' site • connection of sensors and other equipment in a customer LAN	• more mature than ZigBee • many products already available • permits higher data rates than ZigBee	• so far, most equipment does not have Bluetooth implementation • limited maximum number of devices in a network • security vulnerabilities

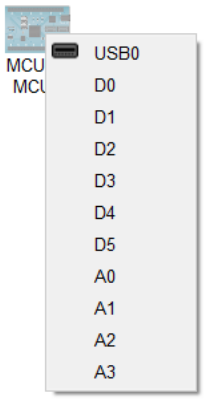
12.3. Последователност на работа и задачи за изпълнение

12.3.1. Управление и мониторинг на IoT устройства – сензори и изпълнителни механизми, с помощта на MCU device board и модулно програмиране

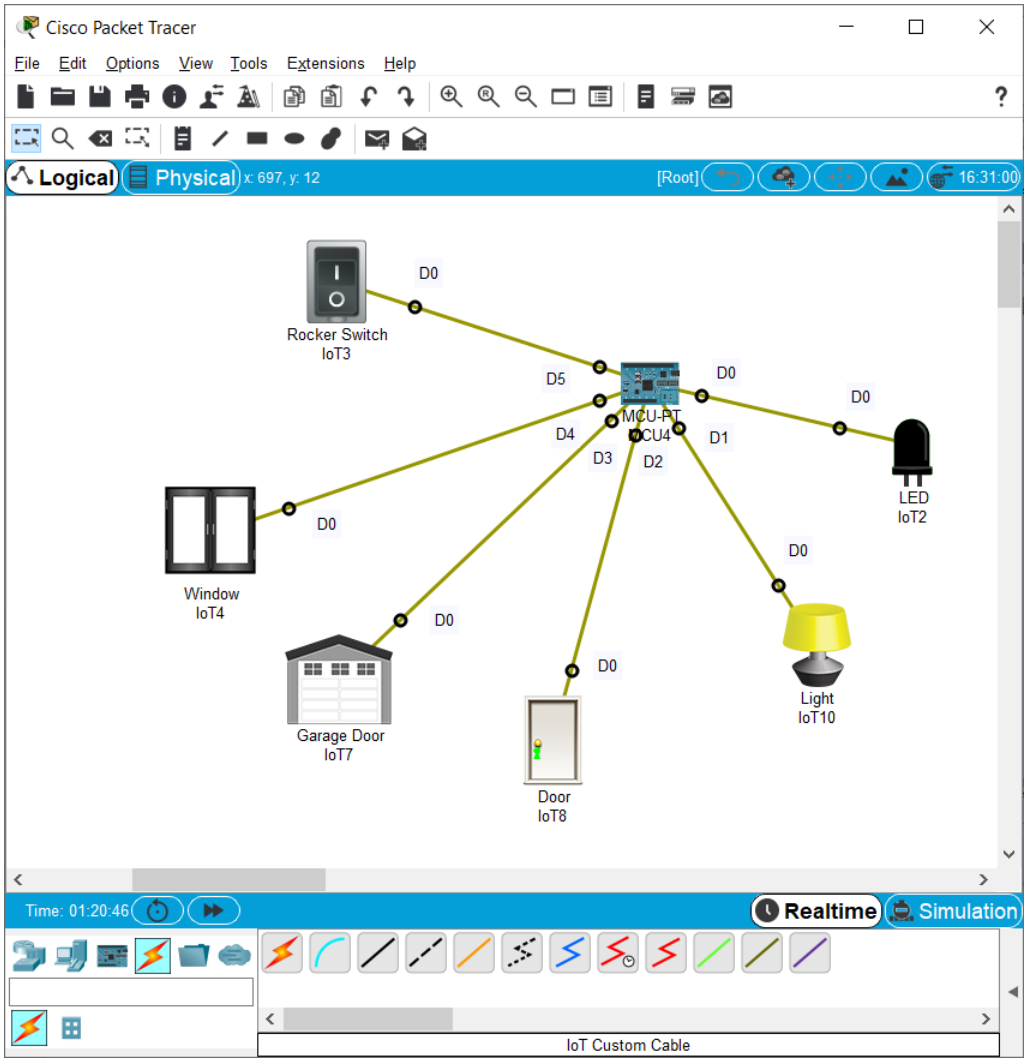
Една от възможностите за свързване и управление на IoT устройства в среда Cisco Packet Tracer се свързва с употреба на многофункционален борд модел MDU-PT, достъпа, до който е показан на фиг. 12.4. Модулът разполага с набор от USB порт, цифрови D0 – D5 и аналогови A0 и A3 входове/изходи (фиг. 12.5) позоляващи достъп и конфигурация на сензорни устройства и изпълнителни механичми при различни реализирани мрежови сценарии.



Фиг. 12.4. *Достъп до MCU Device Board*



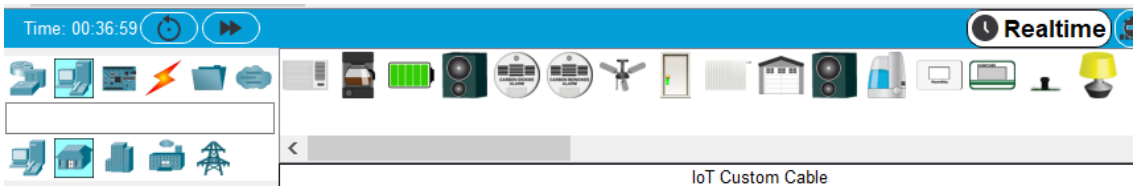
Фиг. 12.5.*Налични портове при MCU Device Board*



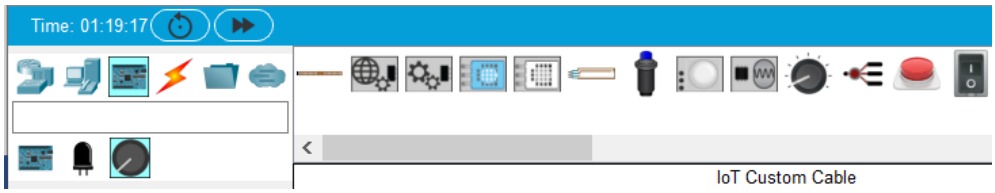
Фиг. 12.6. *Управление на IoT устройства чрез IoT превключвател Rocker Switch при MCU Device Board*

Фигура 12.6 представя мрежова топология на IoT устройства LED (светодиод), Ligth (лампа), Door (входна врата), Garage Door (гаражна врата) и Window (прозорец), свързани посредством и конфигурирани като цифрови изходи на указаните устройства, портове от D0 до D4. Менюто с функционални мрежови инструменти за достъп до целевите IoT и устройството за тяхно управление са показани на фиг. 12.7.

а)



б)

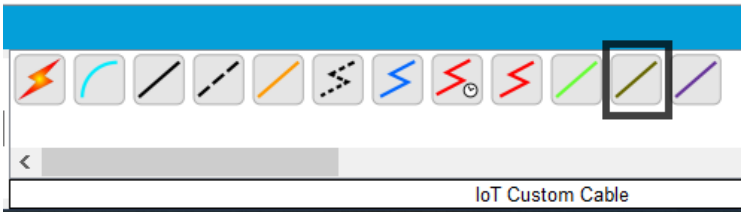


в)

Фиг. 12.7. *Дотъп до IoT устройства а) LED,*
б) „Ligth, Door, Garage Door и Window“ и в) Rocker Switch

Фигура 12.6 представя мрежова топология на IoT устройства LED (светодиод), Ligth (лампа), Door (входна врата), Garage Door (гаражна врата) и Window (прозорец), свързани посредством и конфигурирани като цифрови изходи на указаните устройства, портове от D0 до D4. Менюто с функционални мрежови инструменти за достъп до целевите IoT и устройството за тяхно управление са показани на фиг. 12.7.

Предвидено е процедурите по активиране на съответните IoT устройства да се извършват чрез използване на сензорния превключвател Rocker Switch, свързан към порт D5, който се конфигурира като цифров вход. Свързването на мрежовите сутройства се осъществява чрез кабел тип IoT Custom Cable (фиг. 12.8).

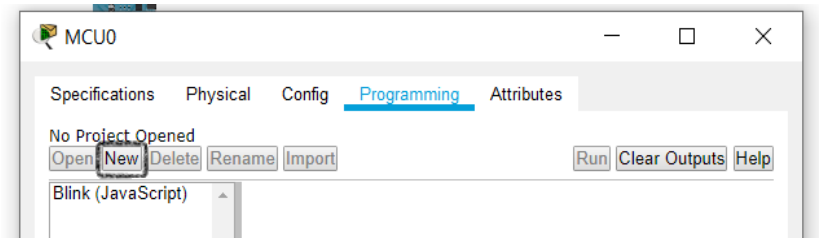


Фиг. 12.8. *IoT Custom Cable*

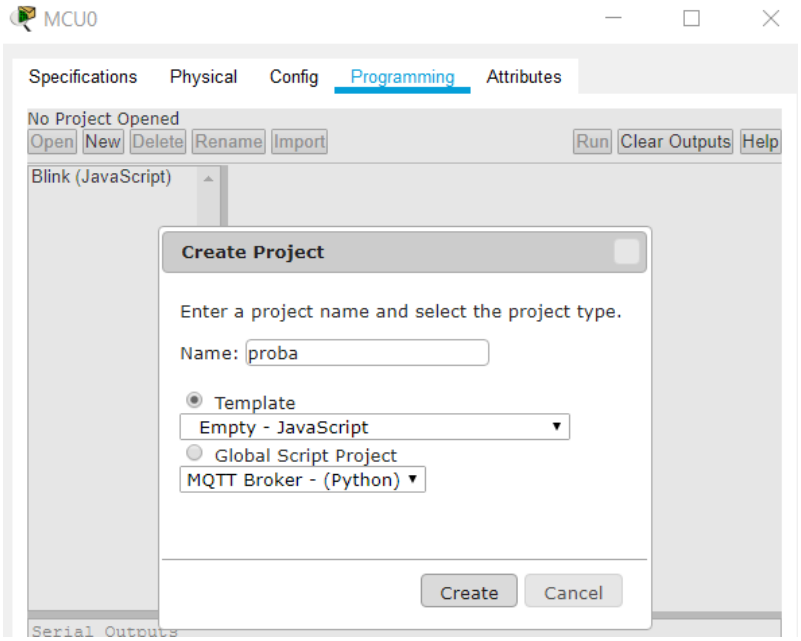
След свързване на мрежовите устройства е необходимо да се проведе програмиране на многофункционалния борд MCU-PT. За целта се избира меню Programming от отворения диалогов конфигурационен прозорец на борда. Програмната настройка на цифровите входове и изходи D0 - D5 може да бъде направена чрез средствата на набор от програмни езици и приложни templates, които могат бъдат дефинирани чрез:

- ❖ избор на нов проект (фиг. 12.9.а));
- ❖ въвеждане на име на проекта и избор на празен програмен template на различни инструменти като JavaScript, Python и Visual, показано на фиг. 12.9.б). За конкретния случай „proba“, „Empty-JavaScript“ и на последно място „create“;

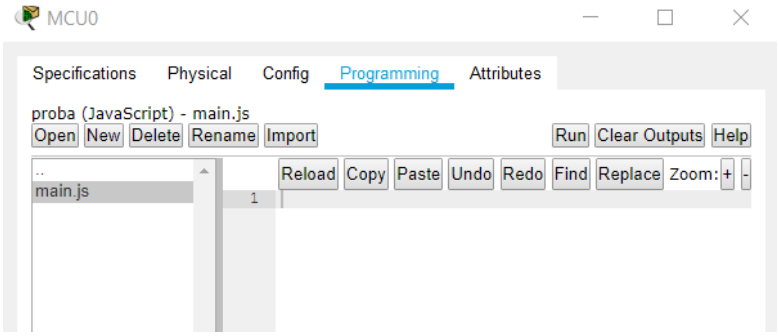
❖ маркиране на основния създаден JavaScript файл „main.js“ и оказване на Open бутон, онагледено на фиг. 12.9.в), след което могат да бъдат извършени стъпки по въвеждане на потребителско програмно съдържание за конфигурация на комуникационните портове.



а)



б)



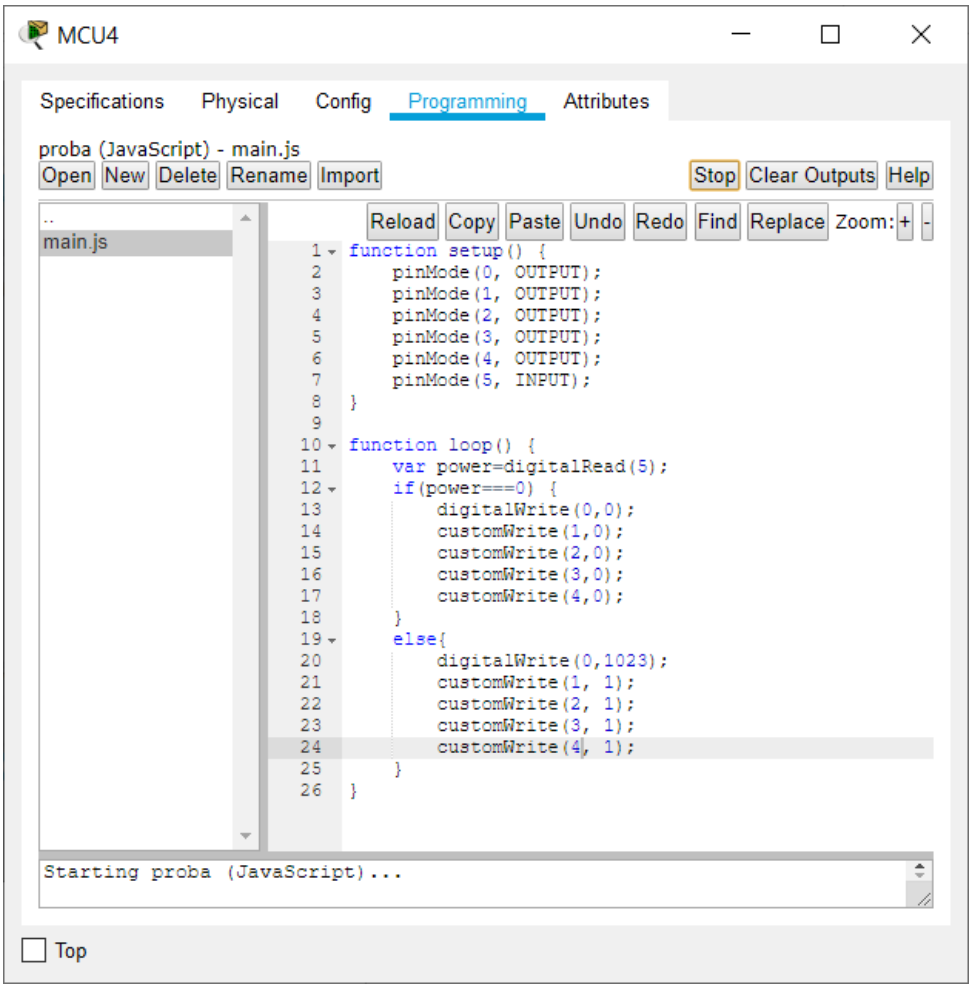
в)

Фиг. 12.9. *Меню Programming а), създаване на нов проект б) и избор на файл за въвеждане на програмно съдържание на модул MCU-PT*

JavaScript програмният код, чрез който се извършва конфигурацията на модула, е показан на фиг. 12.10. Съдържанието на потребителската програма включва две основни функции, както следва:

- function setup {}, чрез която се реализира настройката на цифровите портове D0 – D5 като изходи за IoT устройства LED, Ligth, Door, Garage Door и Window и сензорен вход за управляващия превключвател Rocker Switch;

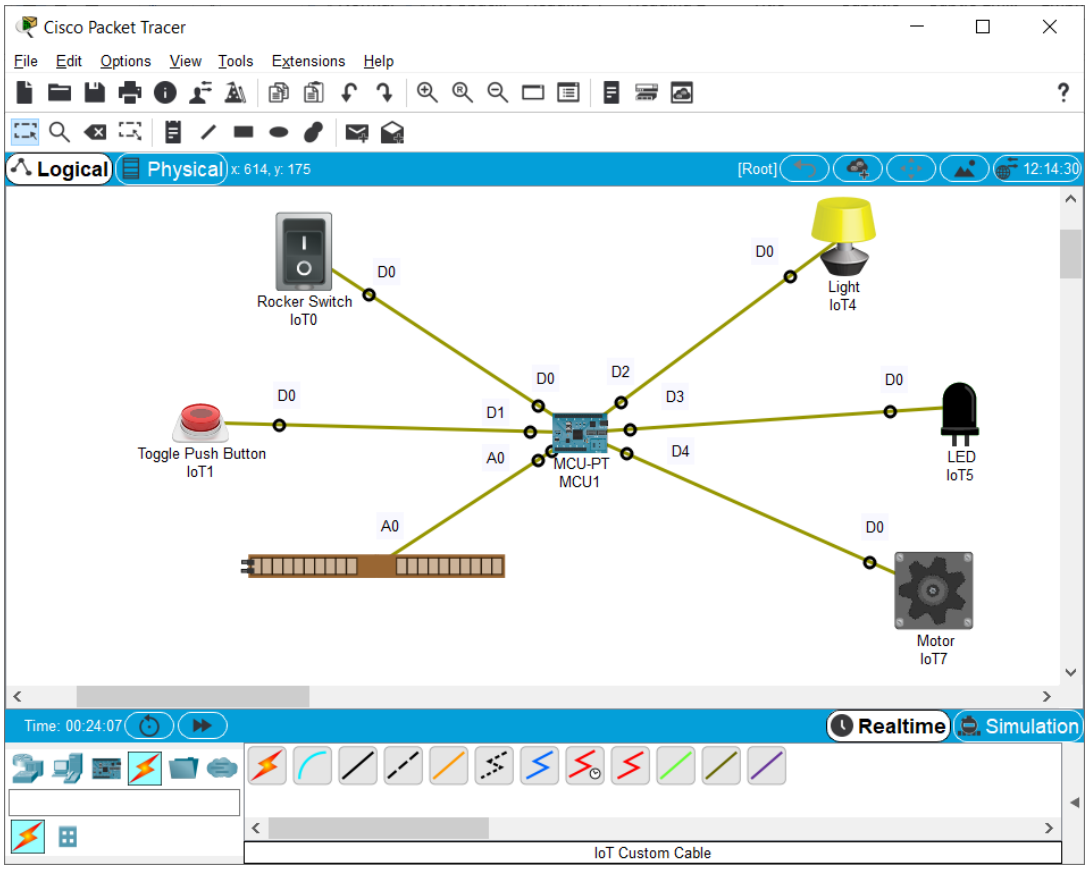
- function loop {} с указване на броя на цифровите портове за следене на състоянието им, както и "if" и "else" програмни сегменти за изключване „0" и задействане „1" на превключвателя и задаване на текущото състояние за неактивност и активация на указаните IoT LED, Ligth, Door, Garage Door и Window устройства, които са обект на мониторинг.



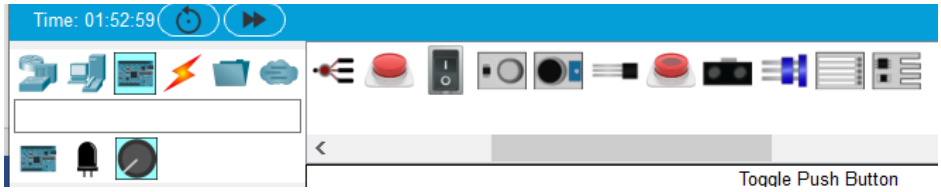
Фиг. 12.10. *JavaScript код за конфигурация на цифровите портове за управление и мониторинг на IoT устройствата*

За тестване на функционалността на системата трябва да се извърши стартиране на въведеото програмно съдържание.

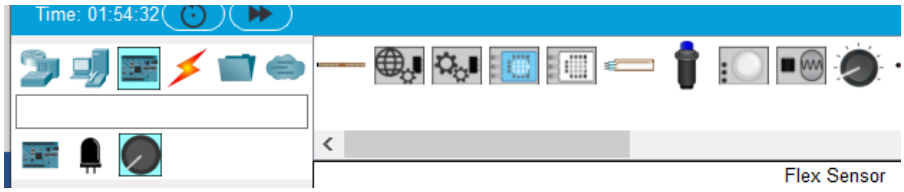
Във връзка с надграждане на обхвата на предствената мрежова топология може да бъде разгледан ***сценарий с включване на „сензорни елементи", на основата на чиято активация, ще бъдат изработвани управляващи въздействия за задействане на набор от съответни „изпълнителни механизми".*** Примерна такава мрежова топология е показана на фиг. 12.11.



Фиг. 12.11. *Топология с включване на IoT сензорни елементи и изпълнителни механизми към многофункционалния модул MCU-PT*



а)



б)

в)

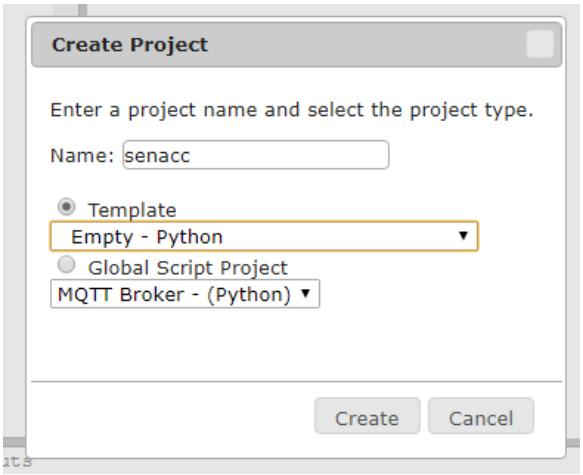
Фиг. 12.12. *Достъп до а) IoT Toggle Push Button, б) IoT Flex Sensor и в) IoT Motor*

Ролята на сензорните елементи и тяхното функционално предназначение е, както следва:

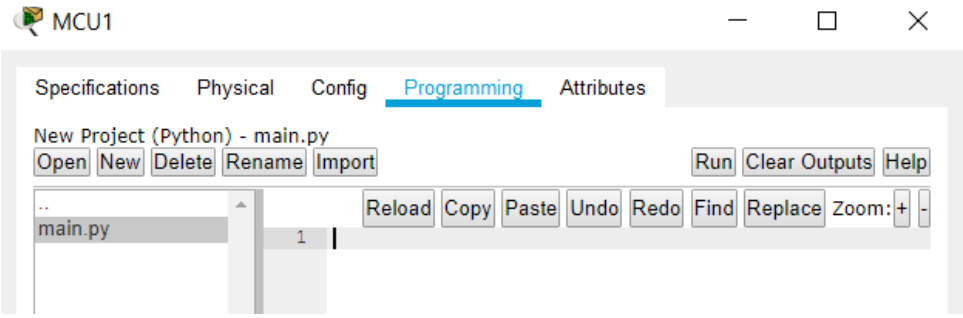
- ❖ сензорен превключвател IoT Rocker Switch (input) - служи за непосредствена активация на нощната лампа IoT Light (output);
- ❖ сензорен бутон IoT Toggle Push Button (input) - използва се за активация на светодиодната IoT LED индикация (output);
- ❖ сензор на огъване IoT Flex Sensor (input) - предсназначен е чрез регулация на неговото положение да забавя или увеличава скоростта на въртене на изпълнителния моторен механизъм IoT Motor (output).

Фигура 12.12 онагледява достъпа до някои от IoT устройсвата, съответно - Toggle Push Button, Flex Sensor и IoT Motor.

За да бъде проведена програмна конфигурация на IoT устройствата на управляващия многофункционален борд, е необходимо да бъде приложена подобна аналогия и да бъде създаден „проект“ с включване на темплейт “Empty Python” (фиг. 12.13). Указаният файлов шаблон “main.py”, показан на фиг. 12.14, ще бъде използван за въвеждане на конфигурационен сорс код, базиран на инструмента Python.



Фиг. 12.13. *Създаване на проект при избор на шаблон Empty – Python*



Фиг. 12.14. *Python шаблон за въвеждане на конфигурационен сорс код за управление на сензорните елементи и изпълнителните устройства*

По-долу е представено описание на необходимото програмно конфигурационно съдържание, въвеждано в структурата на Empty – Python темплейт:

- импортиране на всички системни модули в структурата на GPIO и Time библиотеките;

```
from gpio import *
from time import *
```

- инициализация и присвояване на стойност „0“ на глобални променливи на съответстващи утройства IoT Rocker Switch, IoT Toggle Push Button и IoT Flex Sensor;

```
switchValue = 0
tooglePushButtonValue = 0
flexSensorValue = 0
```

- дефиниране на функция за прочитане на мониторинг на състоянието на сензорните елементи, отнасяща се до:
  - ✓ декларация на променливи на switchValue, tooglePushButtonValue и flexSensorValue на IoT превключвателя, сензорния бутон и сензора за огъване като „глобалми променливи“;
  - ✓ прочитане на състоянието на глобалните променливи switchValue, tooglePushButtonValue и flexSensorValue (цифрови входове D0, D1; аналогов вход A0).

```
def readFromSensors():
    global switchValue
    global tooglePushButtonValue
    global flexSensorValue

    switchValue = digitalRead(0)
    tooglePushButtonValue = digitalRead(1)
    flexSensorValue = analogRead(A0)
```

- дефиниране на функция за изработване/генериране на управляващи въздействия на изпълнителните механизми IoT Light, IoT LED и IoT Motor. Прилагат се следните условия:
  - ✓ „ако IoT Rocker Switch е в положение “HIGH - (включен превключвател)”, то установи устройство IoT Light в състояние “2 – (активиране на висока интензивност на генерираната светлина)”, в противен случай, преустанови активацията на IoT Light или задай състояние “0 – (липса на излъчвана светлина)”;
  - ✓ „ако IoT Toggle Push Button е в активно състояние “HIGH - (натиснат бутон)”, то активирай IoT LED в състояние “ HIGH – интензивно LED лъчение)”, в противен случай, преустанови активността на индикацията IoT LED или фиксирай състояние “LOW – (липса на излъчвана светлина)”;
  - ✓ „ако IoT Flex Sensor е подложен на огъване “стойността на съответстващата му променлива е по-голяма от нула”, то задействай механизъм IoT Motor със скорост на въртене равна на фиксираната от стойността на променливата на сензорния елемент IoT Flex Sensor, в противен случай, преустанови активността на моторния механизъм IoT Motor или задай състояние “0 – (спрям моторен механизъм)”.

```
def writeToActuators():
    if (switchValue == HIGH):
```

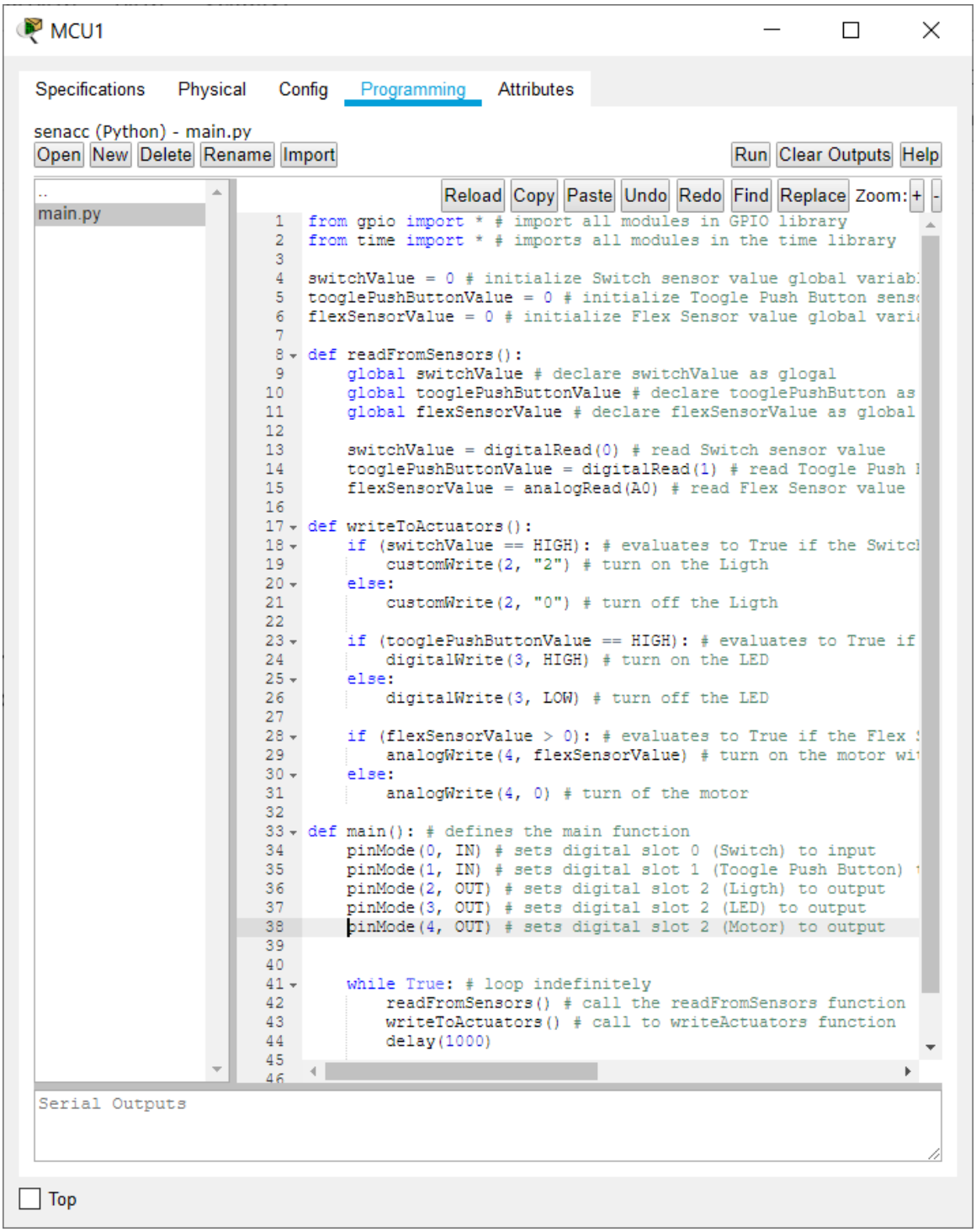
```
            customWrite(2, "2")
        else:
            customWrite(2, "0")

        if (tooglePushButtonValue == HIGH):
            digitalWrite(3, HIGH)
        else:
            digitalWrite(3, LOW)

        if (flexSensorValue > 0):
            analogWrite(4, flexSensorValue)
        else:
            analogWrite(4, 0)
```

- дефиниране на main функцията, обвързана с назначаване на:
  - ✓ цифров вход DO на сензорния превключвател;
  - ✓ цифров вход D1 за сензорния бутон;
  - ✓ цифров изход D2 на нощната лампа;
  - ✓ цифров вход D3 при LED индикацията;
  - ✓ цифров вход D4 във връзка с моторния механизъм;

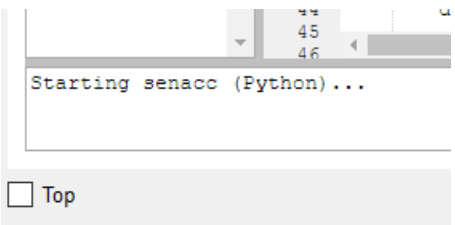
  и организация на циклична структура while, касаеща изпълнението на фунциите за следене на състоянието на сензорните елементи и управление на изпълнителните механизми, докато е налице условие “True”.

```
def main():
    pinMode(0, IN)
    pinMode(1, IN)
    pinMode(2, OUT)
    pinMode(3, OUT)
    pinMode(4, OUT)

    while True: # loop indefinitely
        readFromSensors() # call the readFromSensors function
        writeToActuators() # call to writeActuators function
        delay(1000)

main()
```

Съдържанието на въведения конфигурационен код е показано на фиг. 12.15. Във връзка с активиране и тестване на коректността на функционалните IoT устройства отново трябва да се извърши стартиране на формирания програмен код, както е показано на фиг. 12.16.

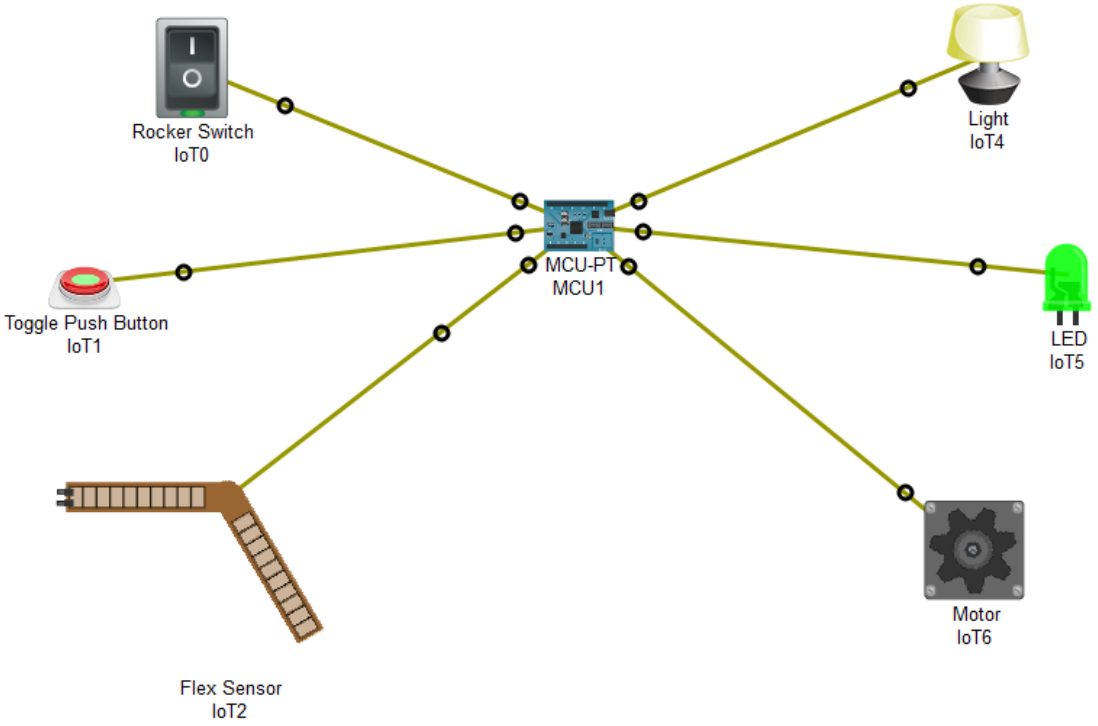


Фиг. 12.15. *Въведен Python програмен код за конфигурация на състоянията на сензорните елементи и изпълнителните устройства*



Фиг. 12.16. *Стартиране Python програмен код за конфигурация на системните устройства*

Фигура 12.17 визуализира действието на конфигурираната мрежова сензорна топология. Активирането на конкретно IoT устройство се постига чрез поставяне на маркера на мишката върху устройството и комбинация от “alt+press”.



Фиг. 12.17. *Стартиране на конфигурацията за мониторинг и управление на IoTсензорни елементи и изпълмителни механизми*

12.3.2. Моделиране на IoT система за температурен мониторинг и контрол в средата на мрежови симулатор CISCO PACKET TRACER

Системният мониторинг на параметри на въздушната среда е важна задача за поддържане на зоната на комфорт в домовете и работните помещения. Във връзка с това се предлагат различни решения от отделни приложения до цялостни системи за сградна автоматизация. Преди да бъдат реално изградени, е добре подобни системи да бъдат компютърно моделирани с помощта на различни софтуерни продукти. Една от възможностите за това се свързва с използването на концепцията IoT в програмната среда Cisco Packet Tracer.

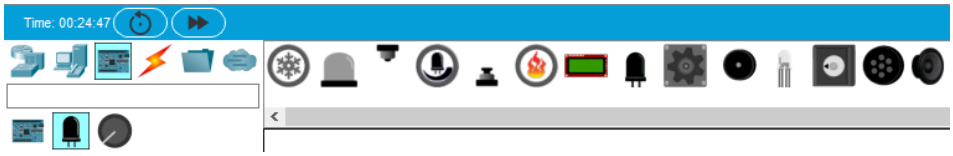
Разглежда се сценарии на примерна мрежова топология на „IoT система за мониторинг и контрол на температура в имитирана реална среда“ в продукта Cisco Packet Tracer с включване на следните компоненти:

- ✓ IoT сензорен елемент за следене на температура „Temperature Sensor“;
- ✓ IoT термоелектрически охладител за понижаване на t°C „Air Cooler“;
- ✓ IoT термоелектрически нагревателен елемент за повишаване на t°C „Heating Element“;
- ✓ IoT LCD дисплей „LCD“ за извеждане на състояние на мониторинг „нормална температура “, „включен охладител“ или „включен нагревател“;
- ✓ Многофункционален сензорен и изпълнителен модул „SBC-PT“, разполагащ с USB интерфейси и Digital I/O;
- ✓ IoT електронно устройство за регистрация и визуализация на текущата измерена температура t°C „Temperature Monitor“;

- ✓ Преносими интелигентни устройства „TabletPC-PT“ и „SMARTPHONE-PT“ за отдалечен мониторинг на контролираната температура на средата,
- ✓ Безжичен рутер „Home Gateway0“, позволяващ формиране на безжична мрежа за достъп до и следене на състоянието на устройството за отдалечен температурен мониторинг от страна на безжичен таблет и смартфон.

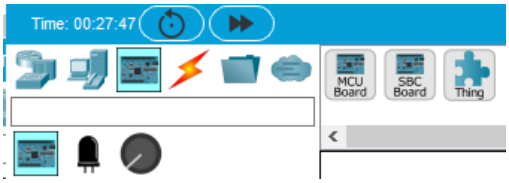
Достъпът до отделните мрежови устройства е онагледен на фиг. 12.18, докато фиг. 12.19 представя компютърно моделираната система за температурен мониторинг в завършен вид.

а)



б)

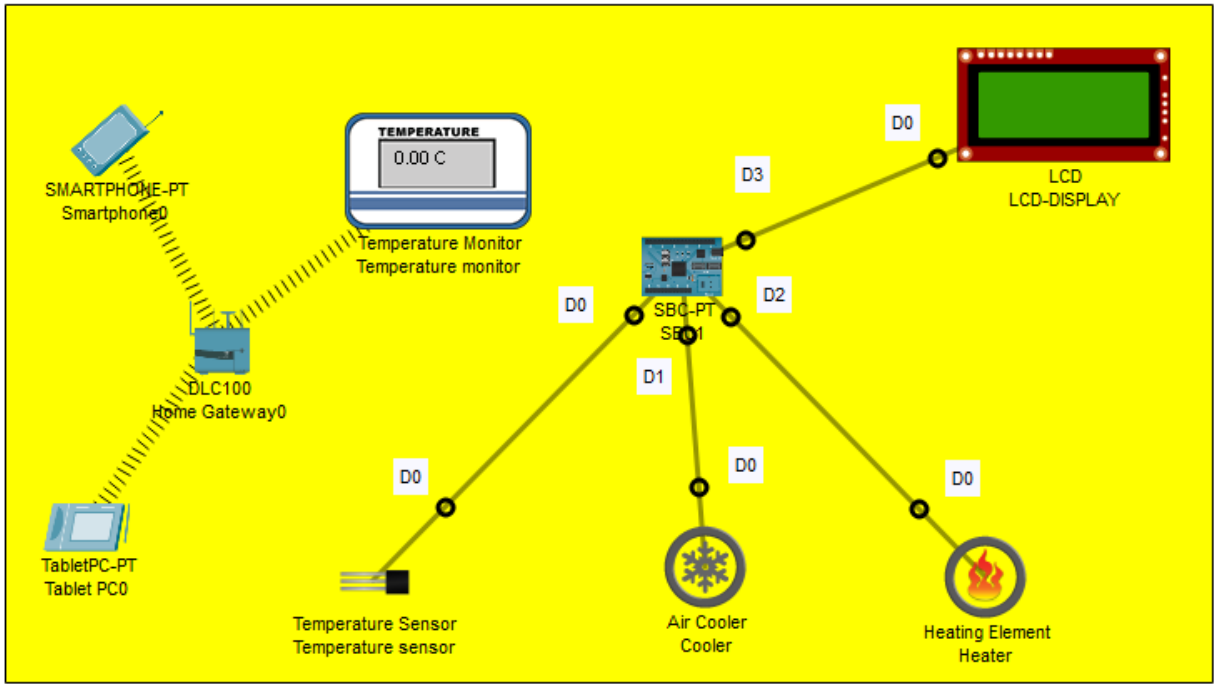
в)



г)

д)

Фиг. 12.18. *Достъп до мрежовите устройства а) „Temperature Sensor“, б) „Air Cooler“, „Heating Element“ и „LCD“, в) „Temperature Monitor“, г) „SBC-PT“ и д) „TabletPC-PT“ и „SMARTPHONE-PT“*

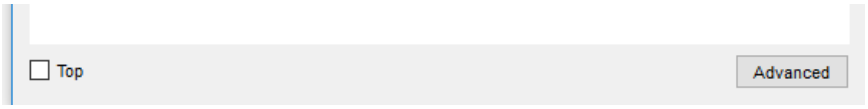


Фиг. 12.19. *IoT система за температурен мониторинг и контрол*

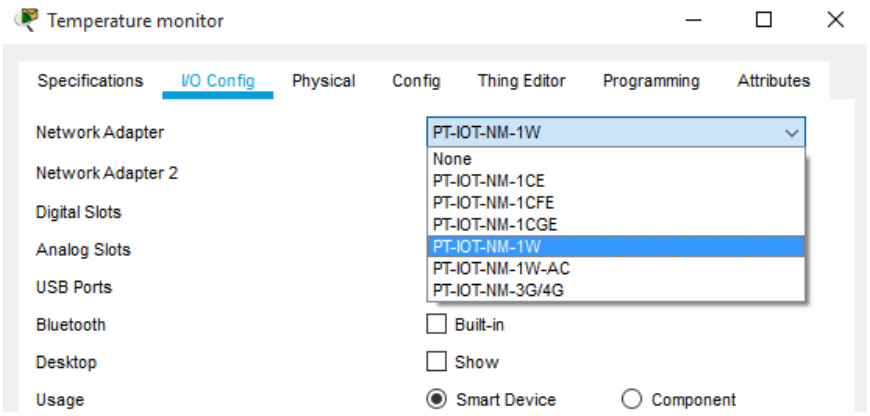
В реализираната мрежова топология са обозначени съответните цифрови интерфейсни I/O за връзка между температурния сензор, охлаждащия и нагревателния елемент, LCD индикацията и многофункционалния борд. Свързването между споменатите IoT устройства се извършва чрез използването на кабел тип "IoT Custom Cable", показан на фиг. 12.20.

Фиг. 12.20. *Свързващ кабел "IoT Custom Cable"*

За свързване на електронното устройство за мониторинг и визуализация на текущата измерена температура „Temperature Monitor" към безжичния мрежови сегмент с включване към рутер Home Gateway0 е необходимо да се извърши избор на конкретен мрежови адаптер. За целта се избира „Advanced" – фиг. 12.21.а), след което „IoT Config" и тип безжичен адаптер „PT-IOT-NM-1W" – фиг. 12.21.б). За да бъде IoT устройството „видимо" или достъпно през WEB от страна на потребителския безжичен рутер и смартфона, трябва да се окаже видът на избрания IoT Server – за конкретния случай Home Gateway, както е показано на фиг. 12.22.

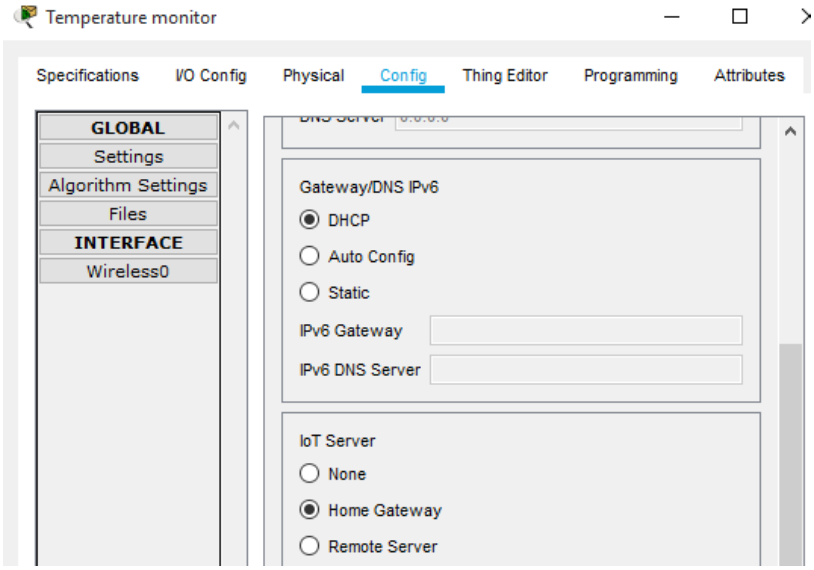


а)

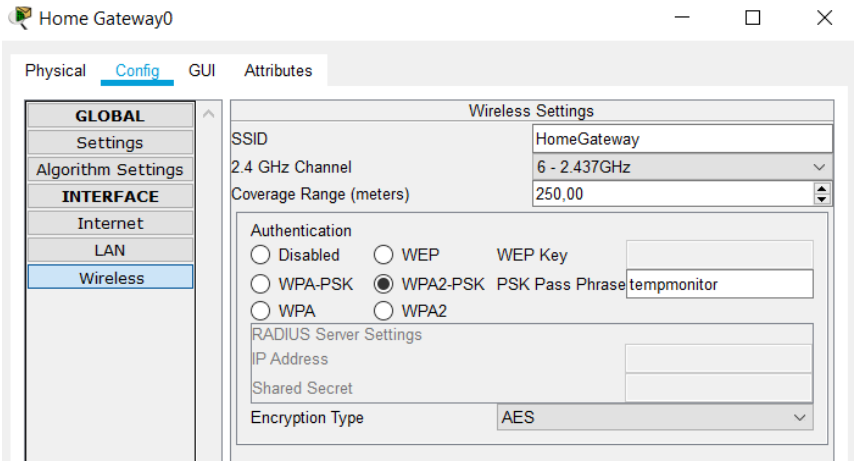


б)

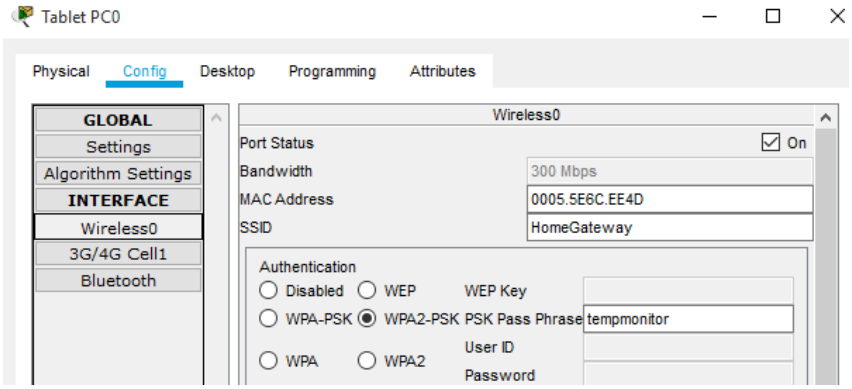
Фиг. 12.21. *Задаване на мрежови адаптер „PT-IOT-NM-1W“ – а) „Advanced“ и б) „IoT Config“*



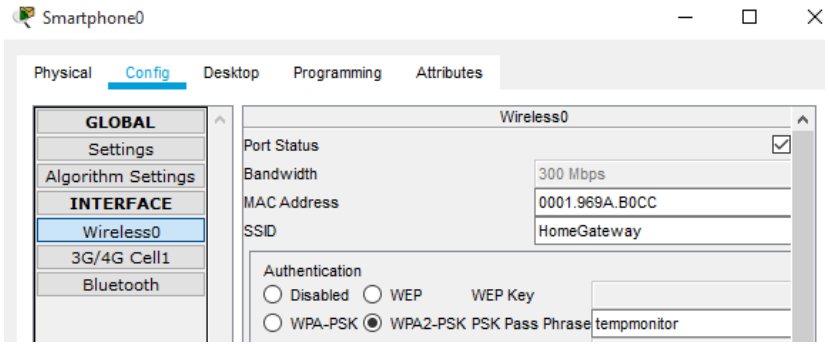
Фиг. 12.22. *Избор на IoT сървър Home Gateway*



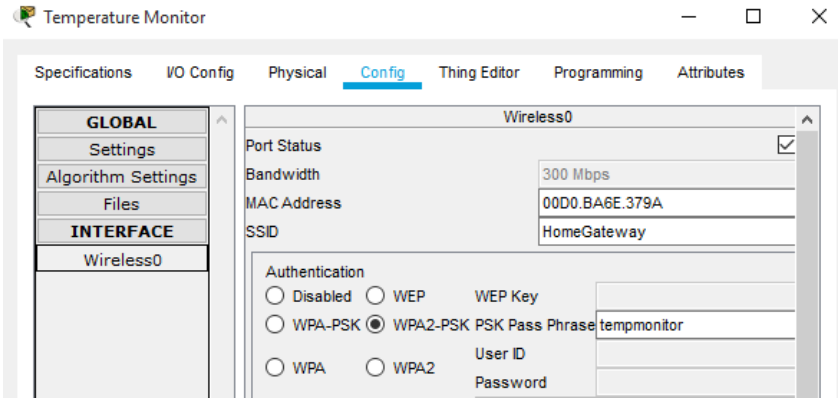
а)



б)



в)



г)

Фиг. 12.23. *Задаване на SSID и режим на автентикация WP2-PSK при интелигентни устройства а) Home Gateway0, б) TabletPC-PT, в) SMARTPHONE-PT и г) Temperature Monitor*

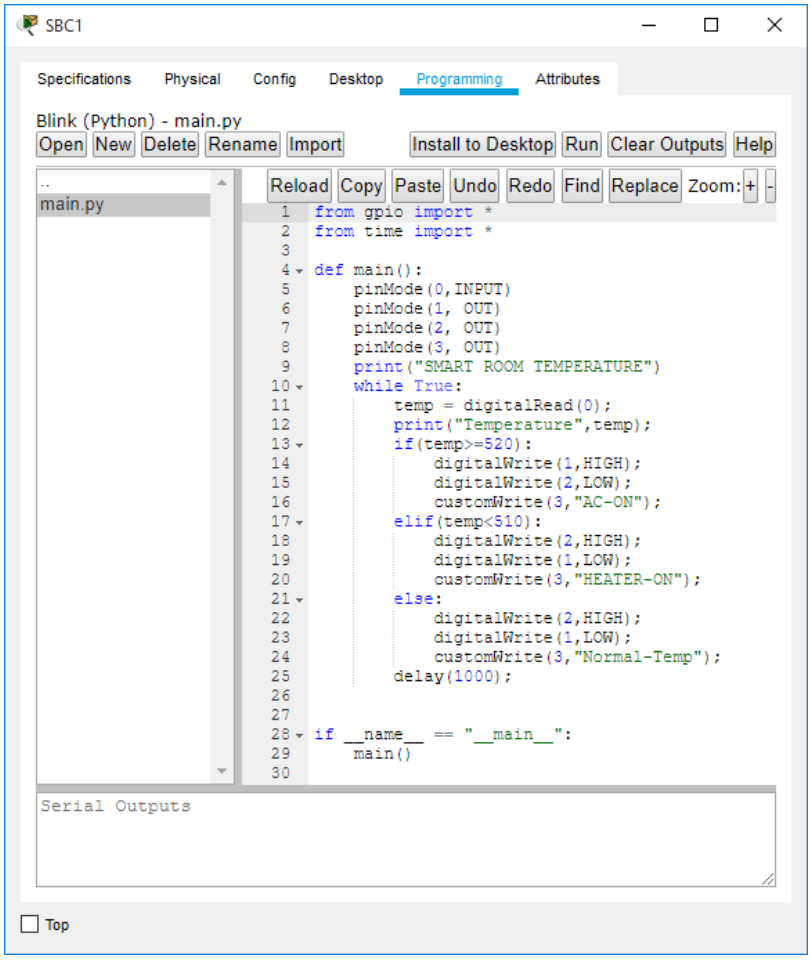
Във връзка със сигурността при безжичния мрежови сегмент, съдържащ устройства Home Gateway0, TabletPC-PT, SMARTPHONE-PT и Temperature Monitor, са приложени, както следва:

- ✓ Унифициран идентификатор на мрежата SSID “HomeGateway”;
- ✓ Режим за автентикация WP2-PSK и предварително споделен ключ с дължина 11 символа „tempmonitor“.

Процедурата, представена на фиг. 12.23, е последователно проведена за комуникационен иинтерфейс „Wireless0“ спрямо указаните потребителски устройства за отдалечен мониторинг.

На следващ етап се преминава към програмиране на многофункционалния сензорен и изпълнителен модул „SBC-PT“ за:

- ✓ следене на температурния сензор;
- ✓ активиране/деактивиране на термоелектрическия охлаждащ и нагревателен елемент;
- ✓ управление на LCD индикацията.



Фиг. 12.24. *Програмен код в SBC-PT модула за управление на IoT системата за температурен мониторинг и контрол*

Тук се използва модифициран „Python“ програмен шаблон „main.py“. Въведеният програмен код, представен на фиг. 12.24, съдържа следните сегменти:

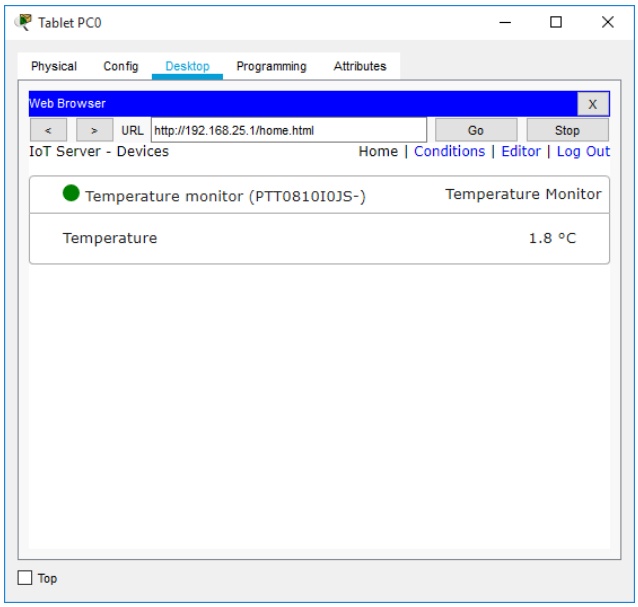
- ✓ модулно импортиране от базисни библиотеки “gpio” и “time”;
- ✓ дефиниране на цифровите входове/ изходи:
  - ✓ Pin ‘0’ към температурния сензор като вход;
  - ✓ Pin ‘1’, ‘2’ и ‘3’ към охладителя, нагревателя и индикацията като изходи;

- ✓ Организация на циклична структура While в три модула с условия съобразно регистрираната количествена температура, предназначени за:

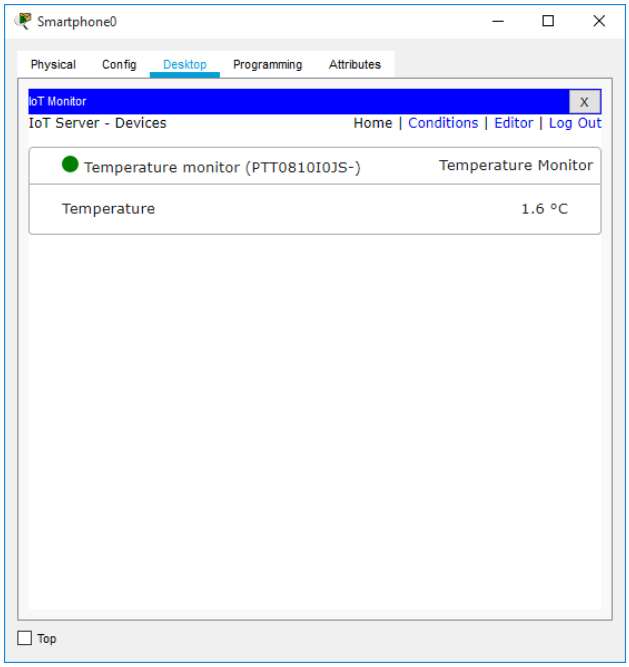
  - ✓ „включване на охлаждащия елемент – състояние HIGH“, „деактивиране на нагревателя – състояние LOW“ и „извеждане на символно съобщение за указване на активен термоелектрически охладител – “AC-ON””;

- ✓ „включване на нагревателния елемент – състояние HIGH“, „деактивиране на действието на охладителя – състояние LOW“ и „извеждане на съобщение за указване на включен термоелектрически нагревателен елемент – “HEATER-ON”“;

- ✓ „поддържане на активността на нагревателя – състояние HIGH“, „поддържане на охладителя в изключено положение – състояние LOW“ и „извеждане на съобщение за указване на моментна нормална регистрирана температура – “Normal-Temp”“.



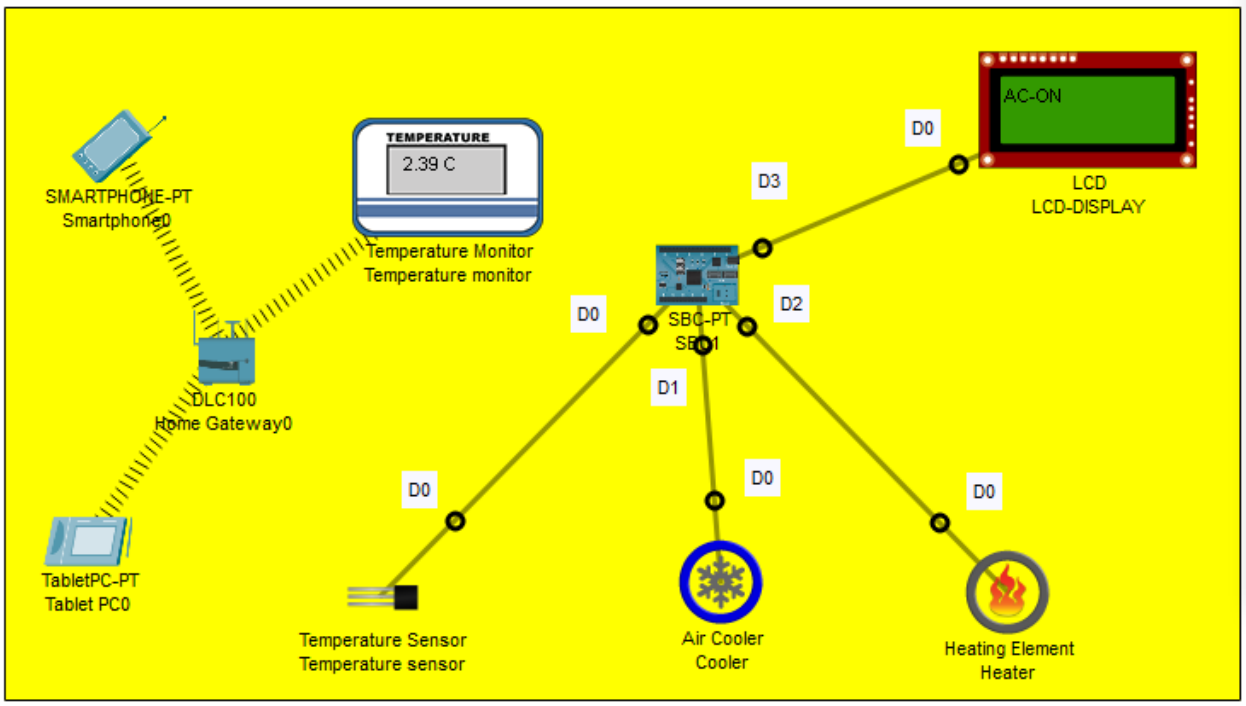
а)



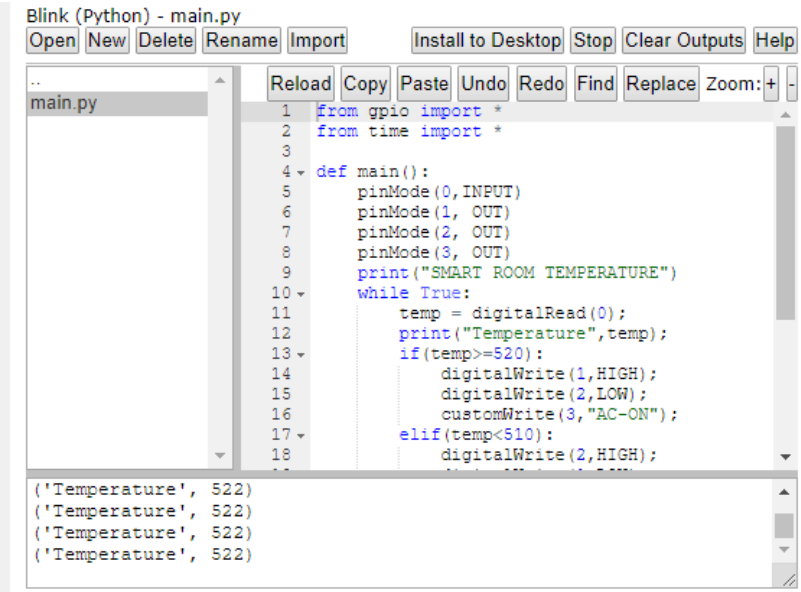
б)

Фиг. 12.25. *Достъп до IoT устройството за температурен мониторинг от а) безжичния таблет през WEB Browser и б) смартфона чрез приложение IoT Monitor*

Достъпът до мрежовото устройство за мониторинг и регистрация на текущата измерена температура в обхвата на обособения безжичен мрежови сегмент може да бъде осъществен чрез използване на WEB Browser или приложение IoT Monitor (фиг. 12.25). Необходимо е да се направи въвеждане/оказване на адреса на безжичния рутер Home Gateway0 „192.168.25.1 – достъпен от LAN интерфейс“, както и потребителско име и парола “admin”.

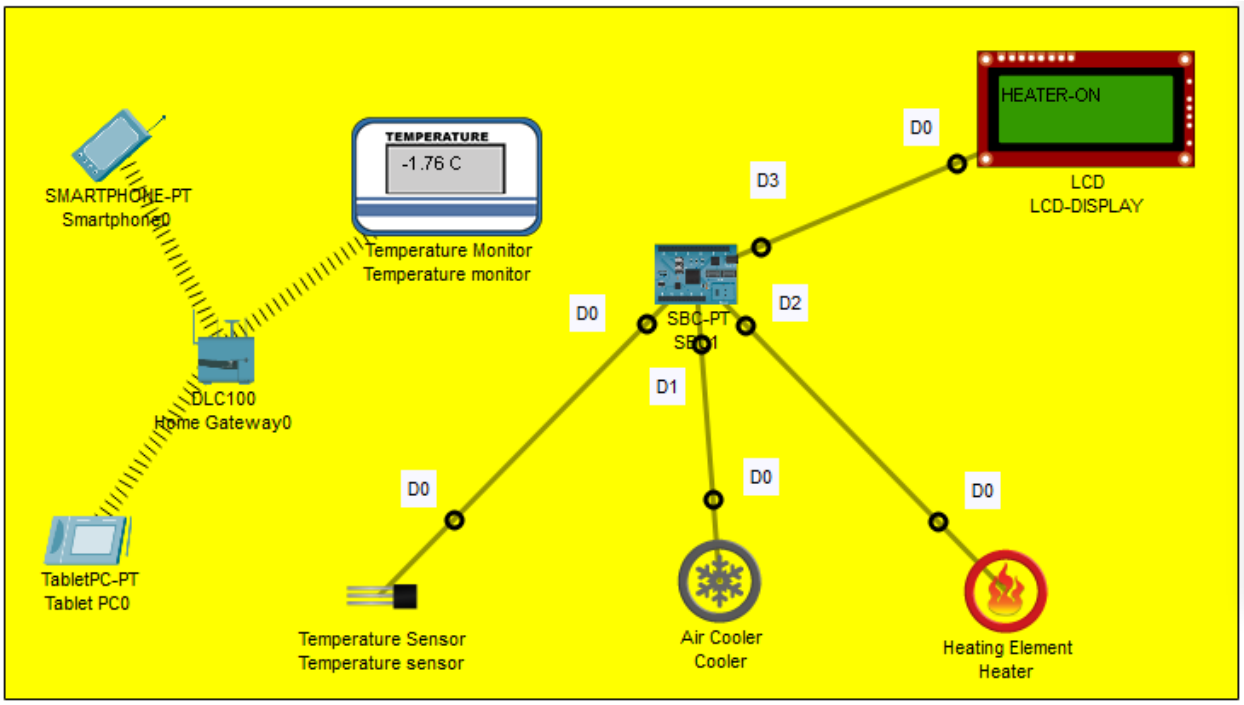


а)

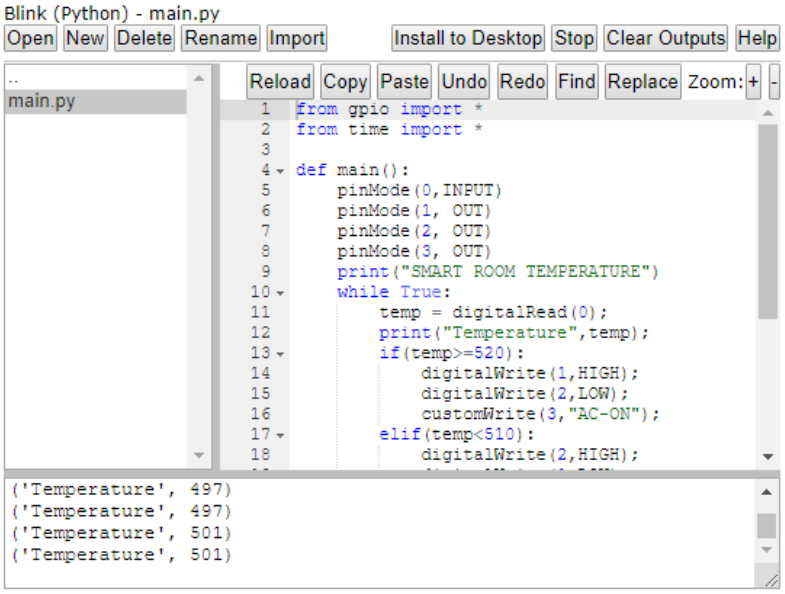


б)

Фиг. 12.26. *Илюстриране на действието на IoT системата за температурен мониторинг и контрол при регистрация на температура над нормалната и включване на термоелектрическия охлаждащ елемент*



а)

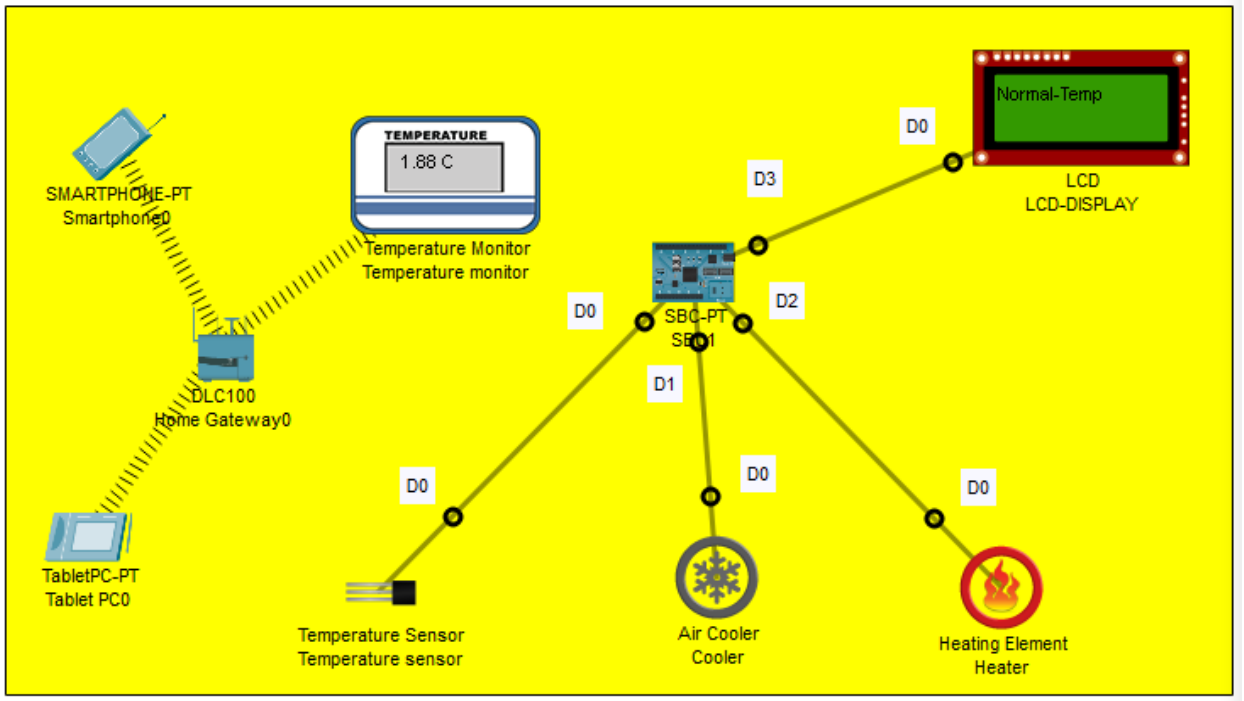


б)

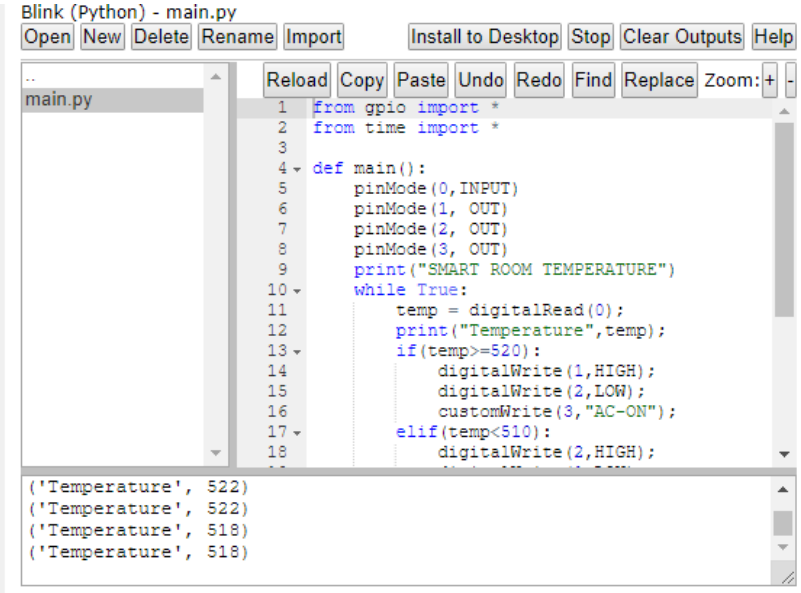
Фиг. 12.27. *Илюстриране на действието на IoT системата за температурен мониторинг и контрол при регистриране на значително понижена температура и включване на термоелектрическия нагревател*

Последната стъпка от тествите процеси представлява верификация на показанията на приборите за регистрация и мониторинг на различните програмно заложени аспекти при установяване на температурни показания:

- ✓ над нормалните – висока t°C;
- ✓ под дефинирания праг за нормална температура - ниска t°C;
- ✓ в приетия диапазон за нормална или средна t°C.



а)



б)

Фиг. 12.28. *Илюстриране на действието на IoT системата за температурен мониторинг и контрол при регистрация на нормална измерена температура*

Функционалността на моделираната IoT система за температурен мониторинг и контрол в основните работни състояния може да бъде проследена от фиг. 12.26 до фиг. 12.28. Непосредствено с това са дадени изгледи на текущи температурни регистрации от изпълнение на отделните програмни сегменти от въведения програмен код в борд SBC-PT.

12.4. Контрролни въпроси.

1. Какви са основните технически дейности при мониторинг на целеви фактори и параметри с използването на сензорни мрежи?
2. Кои са основните структурни компоненти на интелигентните сензорни системи?
3. Конкретизирайте някои от предимствата и недостатъците на WAN технологиите при изграждането на интелигентните сензорни мрежи?
4. Посредством какъв тип интерфейс се създава свързаност между аналоговите / цифровите входове / изходи и IoT сензорите / изпълните механизми?
5. Какви са основните видове шаблони, налични в средата Cisco Packet Tracer, позволяващи структуриране и изпълнение на програмни скриптове за управление на IoT сензорни мрежови конфигурации?
6. Каква е последователността на построяване на логиката на програмните скриптове на моделираните синзорни системи за мониторинг?

Лабораторно упражнение № 13

СИНТЕЗ И МОДУЛНА НАСТРОЙКА НА МРЕЖОВИ КОНФИГУРАЦИИ ИМИТИРАЩИ, IOT СИСТЕМИ ЗА ПОЖАРОИЗВЕСТЯВАНЕ И ПОЖАРОГАСЕНЕ В CISCO PACKET TRACER

13.1. Цел на упражнението

Да се запознаят студентите с техническите средства за следене, регистрация и изработка на управляващи въздействия в моделирани IoT системи за пожарогарене и пожароизвестяване в сумулационна среда Cisco Packet Tracer.

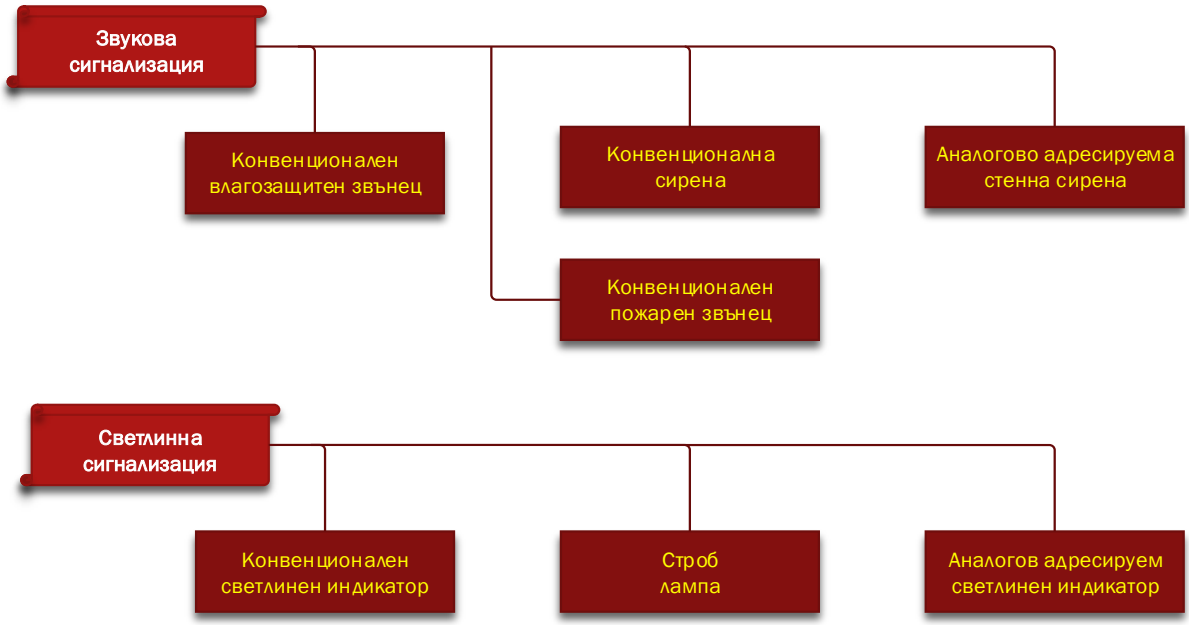
13.2. Теоретична постановка

Пожароизвестителните системи представляват автоматизирани системи за локализация на възникнали пожари в начален стадий и изработване на алармена сигнализация в обособени зони или обекти.

Те се разделят на две основни категории, съответно „конвенционални“ и „аналогово или цифрово адресируеми“.

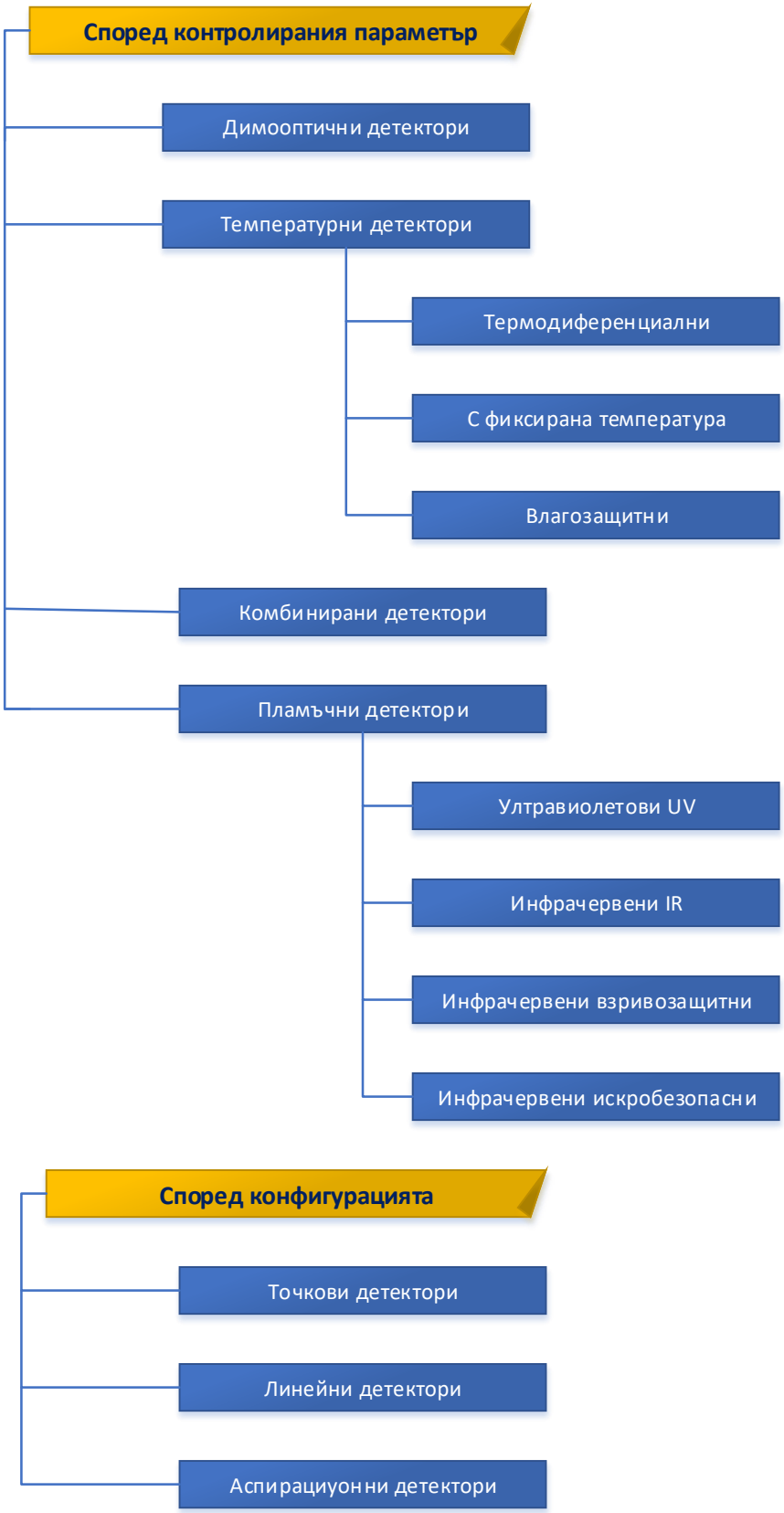
По отношение на вида сигнализация се дефинират две групи технически прибори:

- Устройства за звукова и/или светлинна сигнализация с голяма мощност – при детектиране на възникванал пожар се генерират следните типове индикации „звук“, „светлина“ или „гласови съобщения“;
- Светлинни индикатори, свързани с алармено състояние на конкретен детектор [25, 26].



Фиг. 13.1. *Технически средства за звукова и светлинна сигнализация*

Фигура 13.1 представя основните видове детектори според типа на контролирания параметър и конфигурацията им. На фиг. 13.2 е дадена класификация на устройствата за звукова или светлинна сигнализация.



Фиг. 13.2. *Видове детектори в пожароизвестителните системи*

Съществен аспект при проектирането на модули за пожарогасене в пожароизвестителните и пожарогасителните системи е правилният избор на технически средства за постигане на максимална ефективност при потушаване на възникнали пожари. На фиг. 13.3 са показани основните видове пожароизвестителни модулни системи [25, 26].

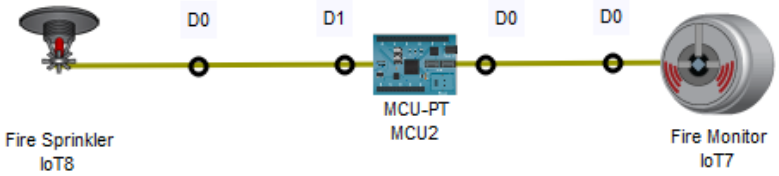
Фиг. 13.3. *Видове пожароизвестителни системи*

13.3. Последователност на работа и задачи за изпълнение

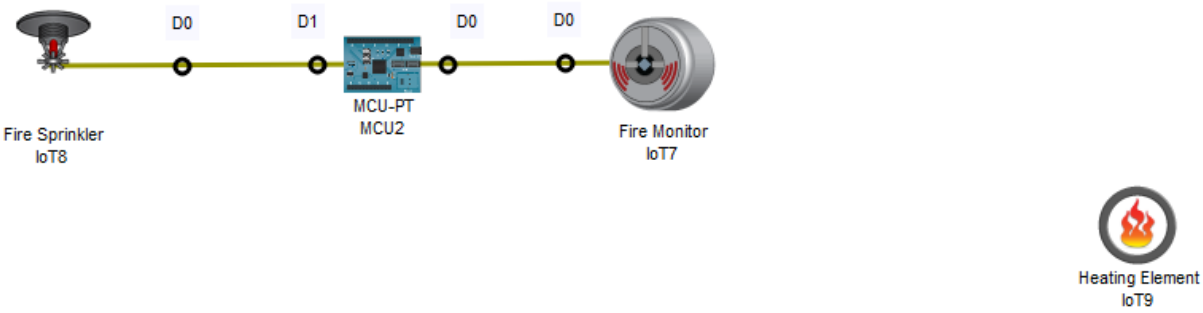
13.3.1. Управление на IoT базиран пожароизвестителен системен модул с включване на MCU модул в Cisco Packet Tracer

Изграждането и конфигурирането на мониторинговата пожарогасителна IoT система преминава в следната последователност:

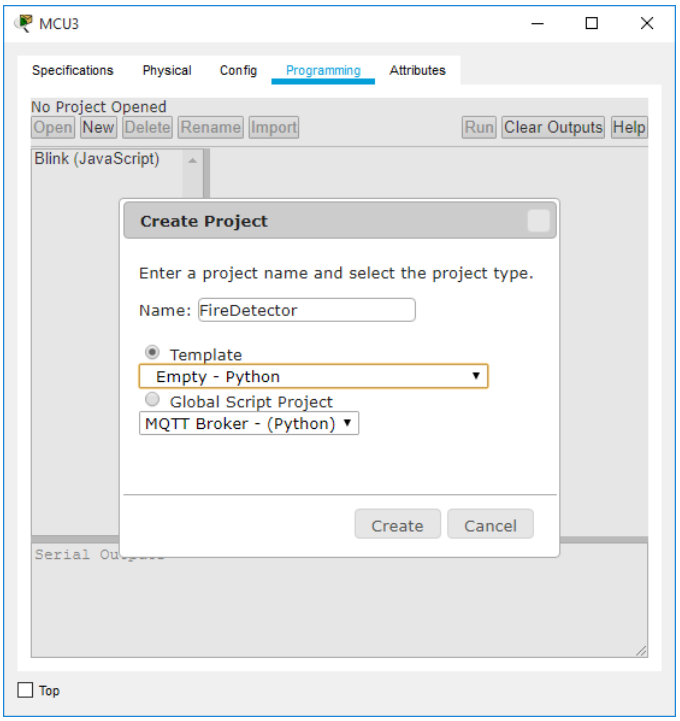
- Свързване на мрежова конфигурация с включване на MCU-PT многофункционален модул; *пламъчен детектор, реагиращ на електромагнитно лъчение в IR спектъра*, IoT Fire Monitor и прибор за гасене на установено възникване на огън IoT Fire Sprinkler (фиг. 13.4). Пламъчният детектор се свързва като сензорния елемент към цифров вход D0, докато IoT устройството за активиране на приток на вода към цифров изход D1 - в ролята на изпълнителен механизъм;



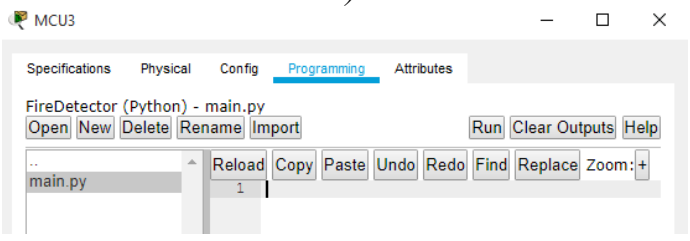
Фиг. 13.4. *Свързване на IoT пламъчен IR детектор и водна пръскачка към борд MCU-PT*



Фиг. 13.5. *Добавяне на IoT Heating Element*



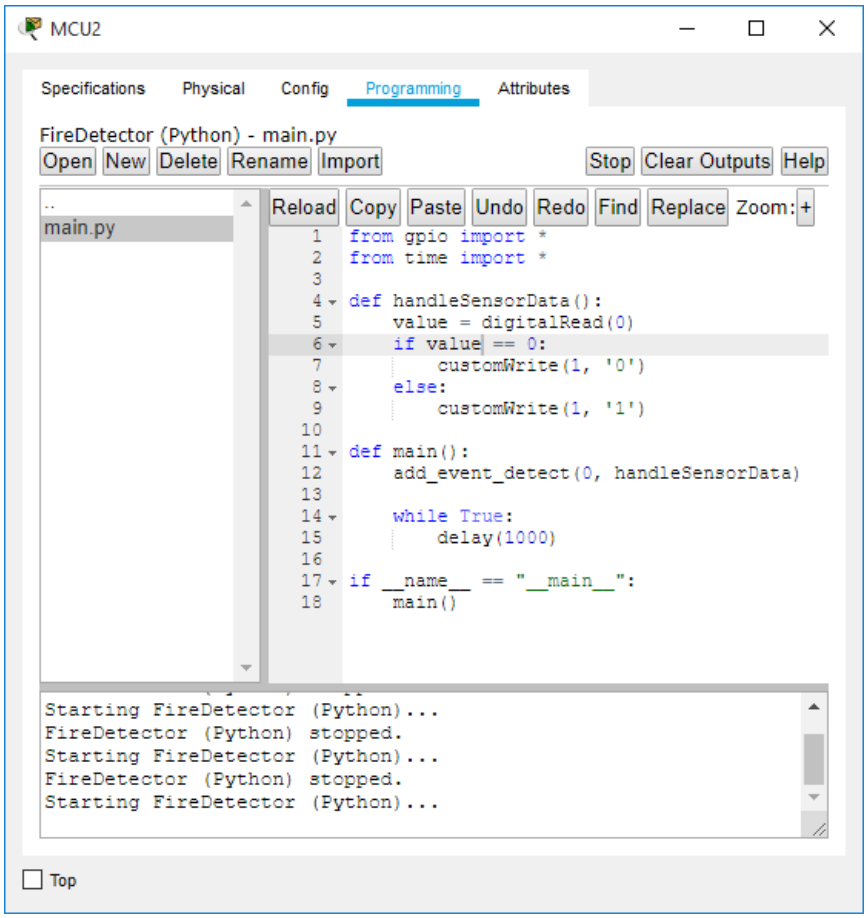
а)



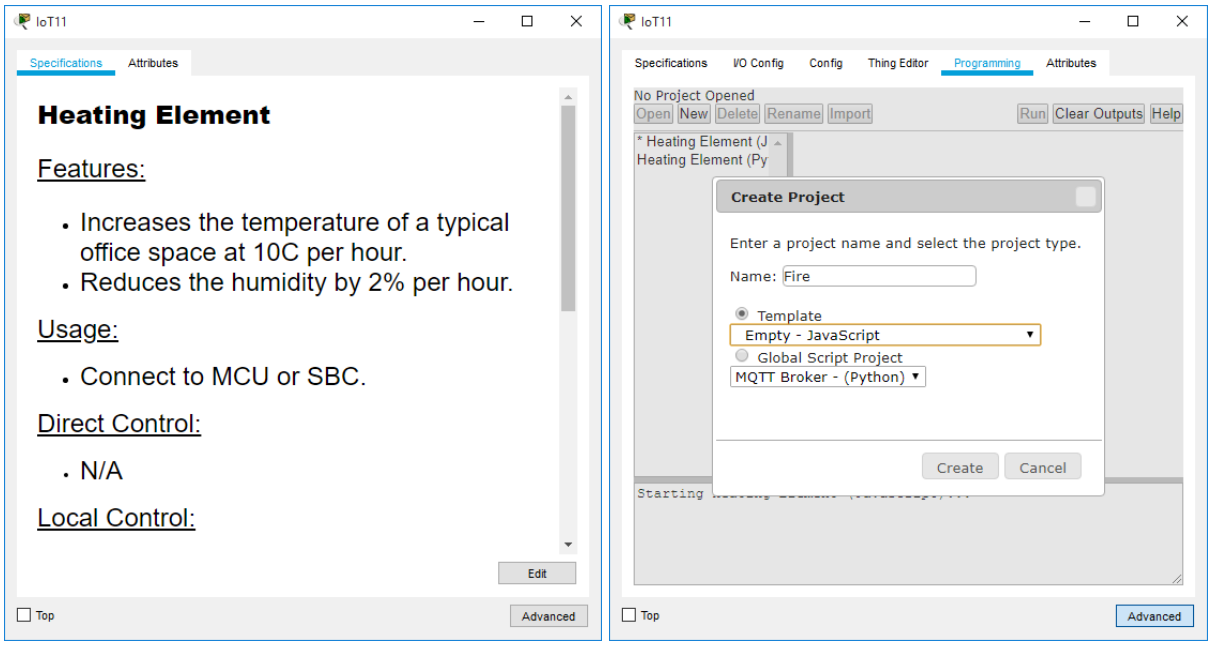
б)

Фиг. 13.6. *Python програмен шаблон – а) създаване и б) поле за въвеждане на сорс код*

- Добавяне на термоелектрически нагревателен елемент IoT Heating Element на известно разстояние по отношение на димния детектор в ролята на източник на топлина за активация на детектора и техническото средство за гасене, показано на фиг. 13.5;
- Програмиране на сензорния модул MCU-PT: Процесът се състои в създаване на Python програмен шаблон и въвеждане на програмни сегменти за включване на специализирани библиотеки и функции, свързани с управлението на IoT устройствата, както е представено на фиг. 13.6 и фиг. 13.7. Програмният код се отнася до оказване достъп за използване на всички “time” и “gpio” входно/изходни функции. Посредством дефиниране на потребителска функция “handleSensorData” се следи цифров вход D0. При наличие на състояние “0 - липса на огън или електромагнитно излъчване в околността на пламъчния детектор” устройство IoT Fire Sprinkler остава в режим на готовност. При състояние “1 – констатирано присъствие на огън или установено лъчение в обхвата на детектора” се извършва програмно задействане на водната пръскачка. Създаването на имитирано събитие за детектиране на присъствието на IR излъчване или огън, изпълнявано при истинност за съответно фиксирано времезакъснение, се извършва чрез функция “main”. Ролята на оператора „if __name__ == "__main__":“ се изразява в оказване на изпълнението на кода вътре в оператор “if” като самостоятелна скриптова програма, която може да бъде импортирана в други програми.

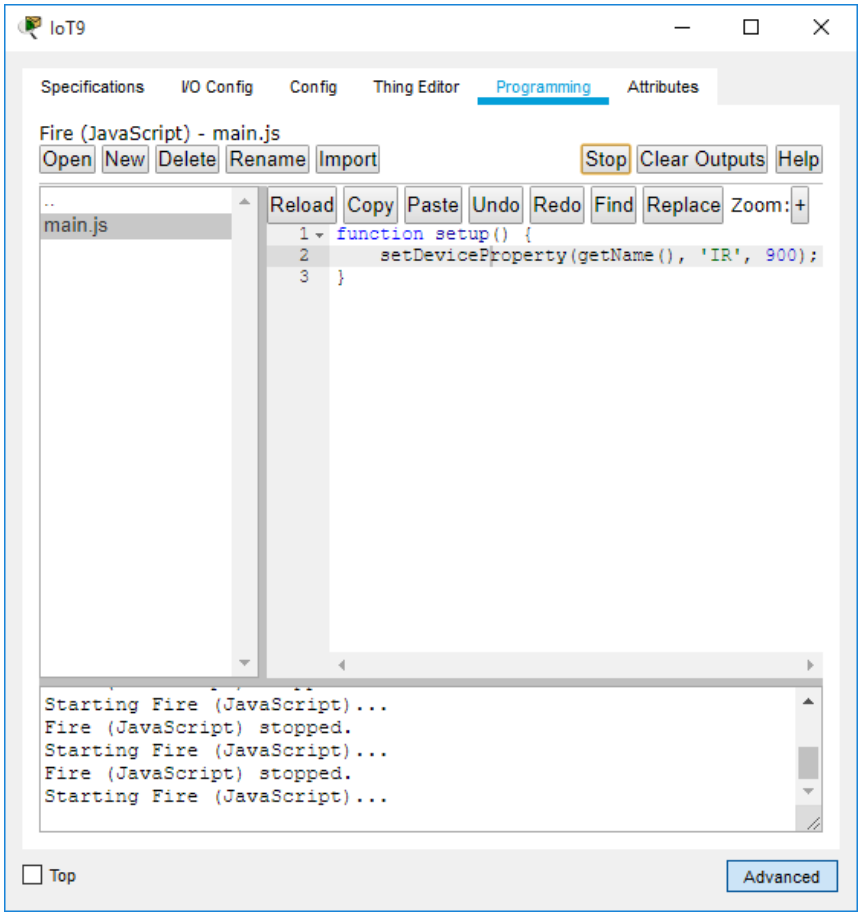


Фиг. 13.7. *Функционален Python програмен код за контрол на сензорния елемент и изпълнителния механизъм*



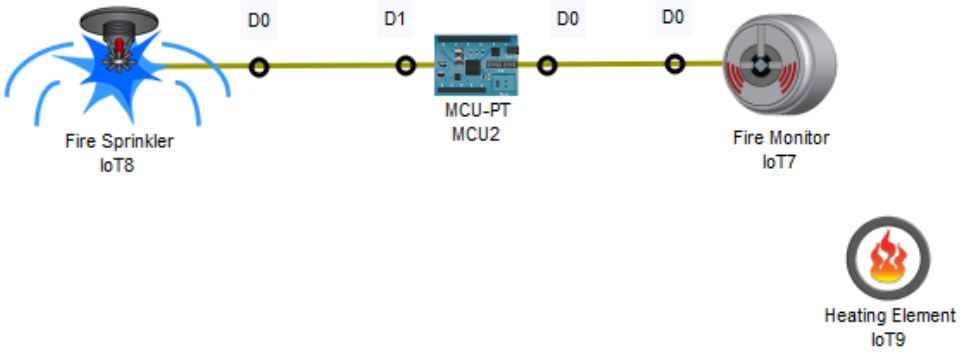
а) *б)*

Фиг. 13.8. *Достъп до менюто "Programmning" а) и създаване на потребителски JavaScript шаблон б)*

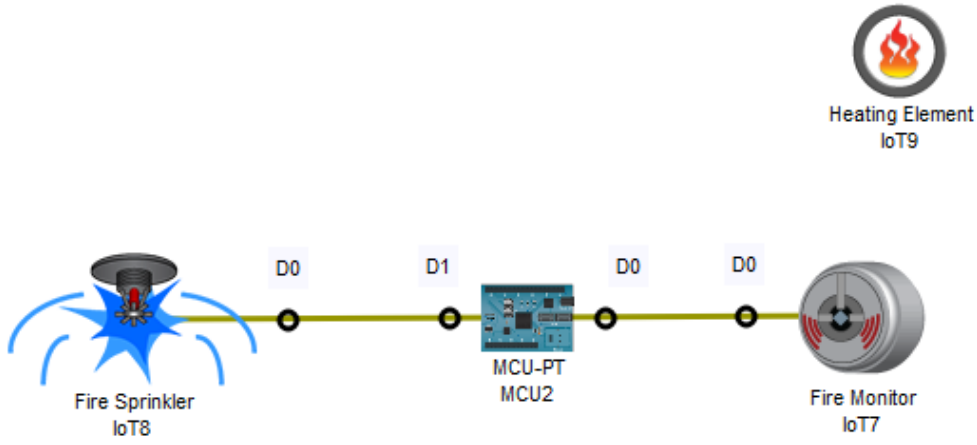


Фиг. 13.9. *Параметрична унификация на IoT Heating Element в JavaScript шаблона*

❖ Програмиране на термоелектрическия нагревателен елемент IoT Eeating Element: Използва се Java Script програмен шаблон като бъде следвана подобна аналогия, както в предходната процедура при езика Python. Процесът на създаване на ".js" шаблона е илюстриран на фиг. 13.8 – избира се "Advanced", с което допълнителните менюта с опции за настройка, включително и "Programming", стават видими. След генериране на Java Script шаблона се въвежда потребителска функция "setup", служеща за параметрично унифициране на текущото IoT устройство на фиг. 13.9;



а)



б)

Фиг. 13.10. *Тестване на мониторинговата пожароизвестителна система с IR пламъчният детектор и термоелектрическият нагревател – първи а) и втори изглед б)*

❖ Тестване на функционалната работоспособност на пожароизвестителната IoT система чрез придвижване на термоелектрическия нагревател в зоната на обхват на пламъчния детектор, функциониращ в електромагнитният инфрачервен спектър, както е показано на фиг. 13.10.

13.3.2. Моделиране на система за превенция от пожари от тип „интелигентен дом" на база на IoT технология в Cisco Packet Tracer

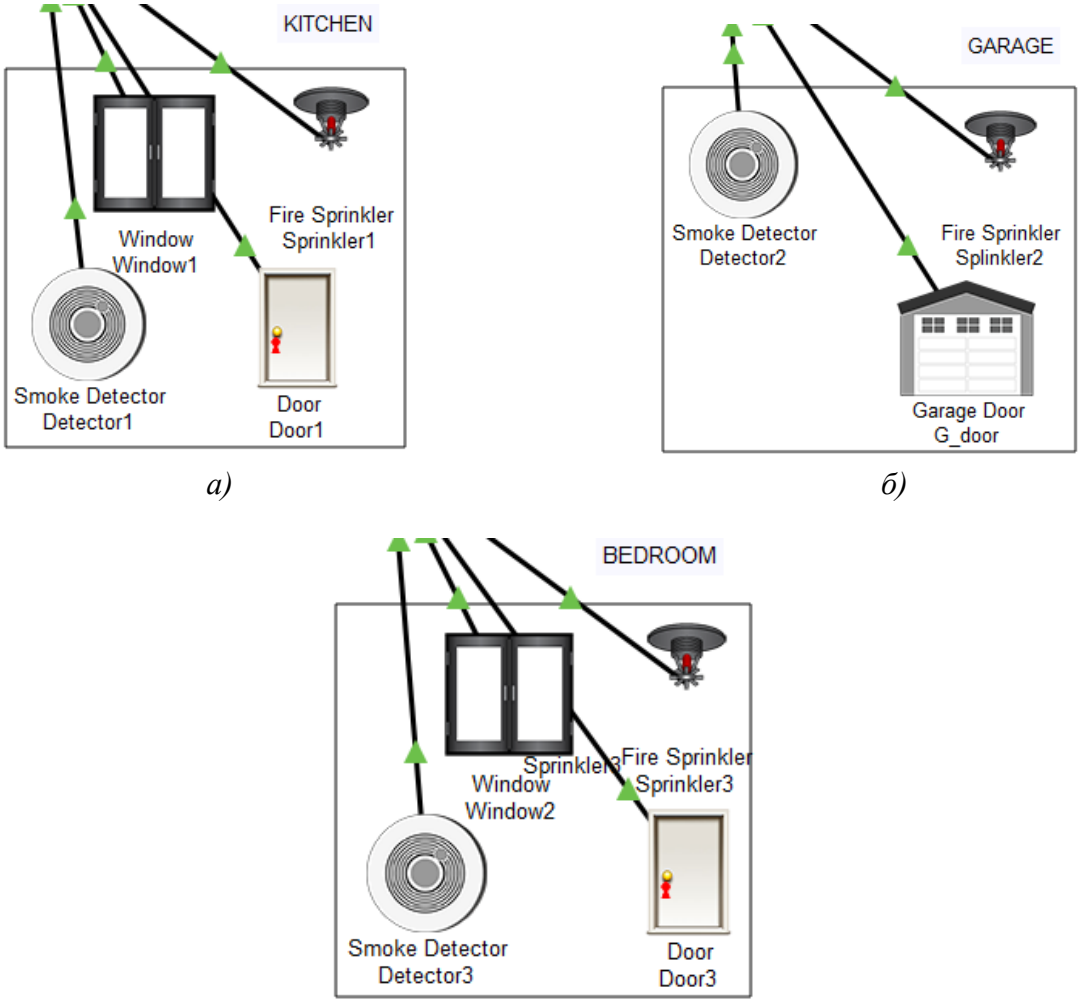
Разглежда се задача за мрежово изграждане, конфигурация и тестване на „***Система Интелиген дом за пожароизвестяване и пожарогасене на база на IoT технология***" в

симулационна среда. Проектирането и тестването на системата преминава през следната последователност от дейности:

- Обособяване на зони, имитиращи стандартни помещения в един съвременен дом като такива могат да бъдат Kitchen, Garage и Bedroom;

- „Инсталиране“ на IoT устройства за детектиране на наличието на дим - Smoke Detecto, и пожарогасене – противопожарна пръскачка Splinkler, в съответните зони. Също така „монтиране“ IoT устройства, врати и прозорци - Windoow, Door и Garage door, както е показано на фиг. 13.11.;

- Въвеждане на IoT устройство с функция за пожароизвестяване със звукова сигнализация – Siren (фиг. 13.12);

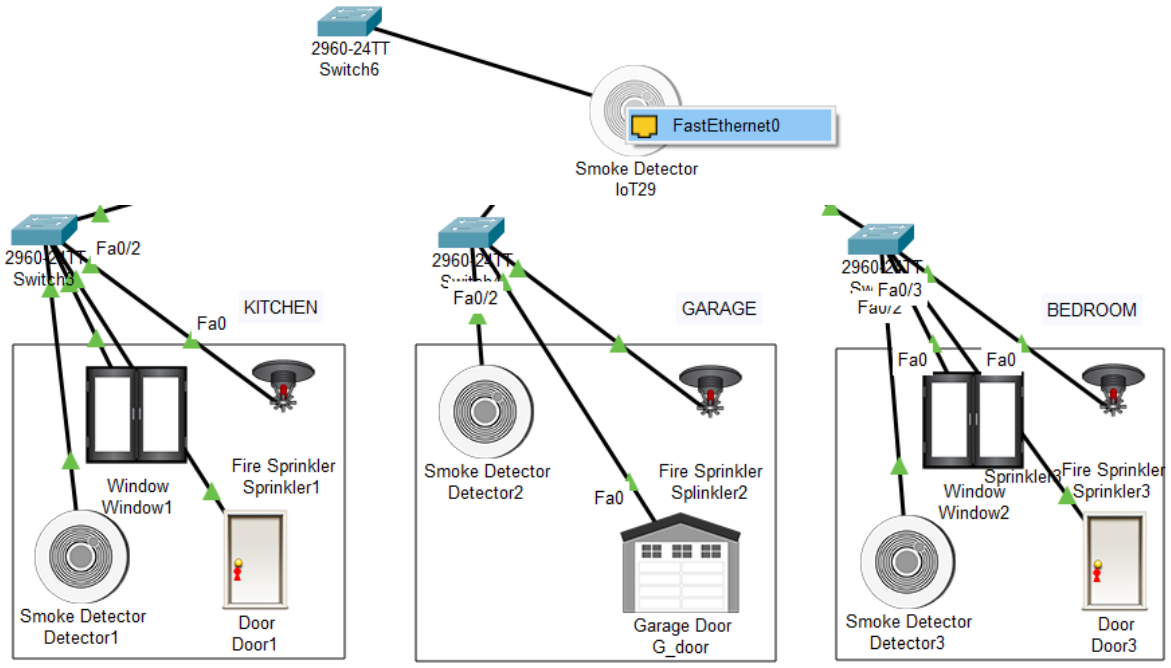


а) *б)* *в)*

Фиг. 13.11. *IoT устройства в помещения Kitchen, Garage и Bedroom*

Фиг. 13.12. *IoT устройство за звукова сигнализация при възникване на пожар*

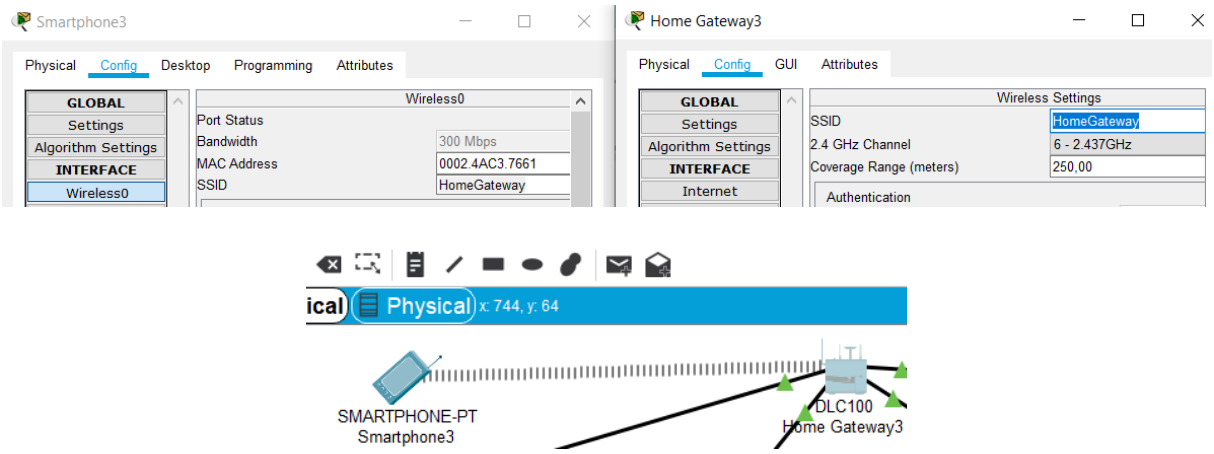
❖ Добавяне на мрежови комутатори Switch модел 2960-24TT за връзка с IoT устройствата от обособените помещения. Свързването на устройствата се реализира по стандартни Fast Ethernet интерфейси чрез връзки тип Copper Straight-Through, както е представено на фиг. 13.13;



Фиг. 13.13. *Свързване на IoT устройствата към мрежовите комутатори*

❖ Въвеждане към топологията на мрежови маршрутизатор DLC Home Gateway и интелигентно преносимо устройство за достъп и управление на активността на всяко от устройствата в системата SmartPhone-PT;

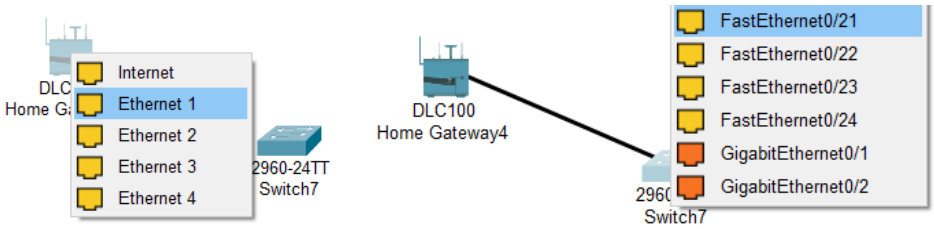
❖ Конфигуриране на безжична мрежова връзка между устройствата SmartPhone-PT и DLC Home Gateway на идентично име на мрежата или SSID (Service Set Identifier), илюстрирано на фиг. 13.14;



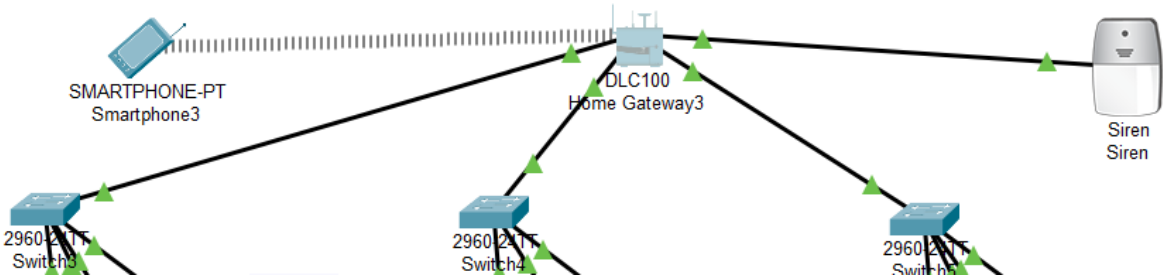
Фиг. 13.14. *Създаване на безжична връзка между Smart phone и Home Gateway*

❖ Свързване на маршрутизатора Home Gateway към IoT устройството за звуково пожароизвестяване Siren и мрежовите комутатори Switchs. Тук е необходимо да се осъществи връзка между Ethernet от страна Home Gateway и Fast Ethernet

интерфейси от страна на указаните мрежови устройства, както е показано на фиг. 13.15 и фиг. 13.16;

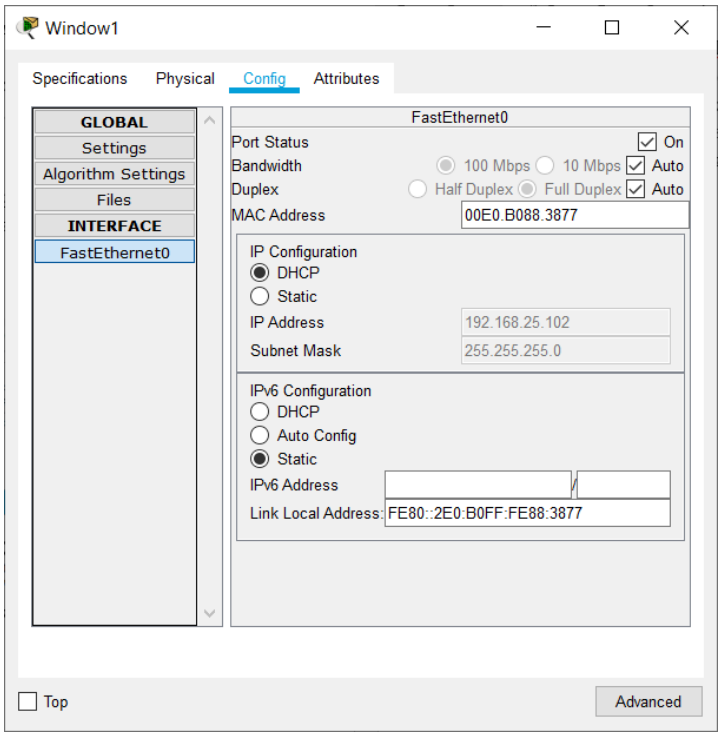


Фиг. 13.15. *Примерно свързване между Home Gateway и Switches*

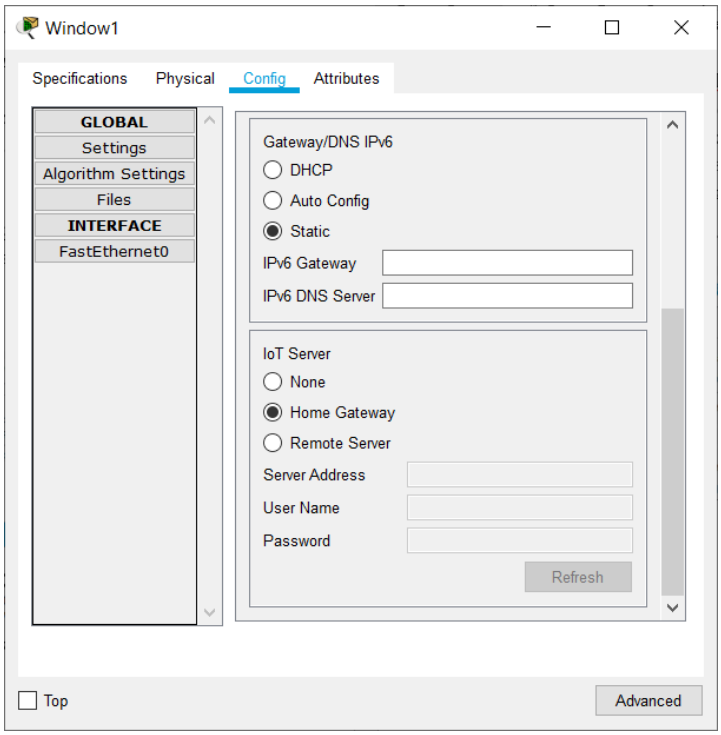


Фиг. 13.16. *Краен вид на свързването между Home Gateway, Siren и Switches*

- Последователна настройка на IoT Server вид Home Gateway и опция за динамично назначаване на IP адреси DHCP относно Fast Ethernet интерфейсите за всяко от IoT устройствата, които са необходими за тяхната *видимост* и възможност за конфигуриране и мониторинг посредством приложение на преносимото устройство Smart Phone, както е онагледено на фиг. 13.17;



а)

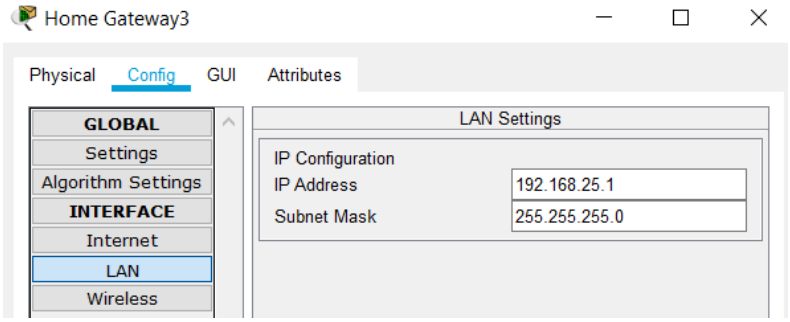


б)

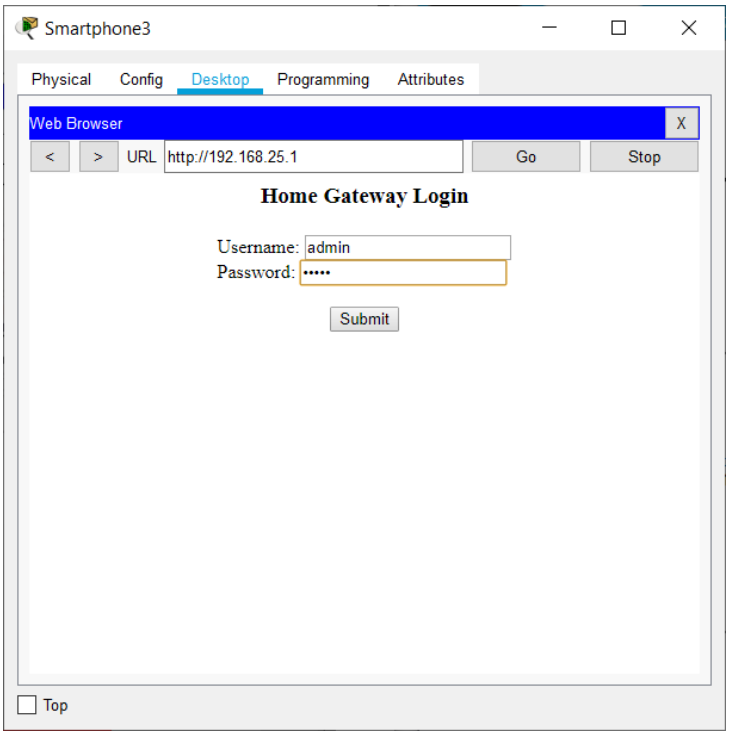
Фиг. 13.17. *Избор на режим на назначаване на а) IP адрес и б) тип IoT сървър*

*<u>забележка:</u> Фигура 13.17 показва настройката на споменатите мрежови параметри само за едно от IoT устройсвата. При всички останали устройства конфигурацията се извършва по аналогичен начин.

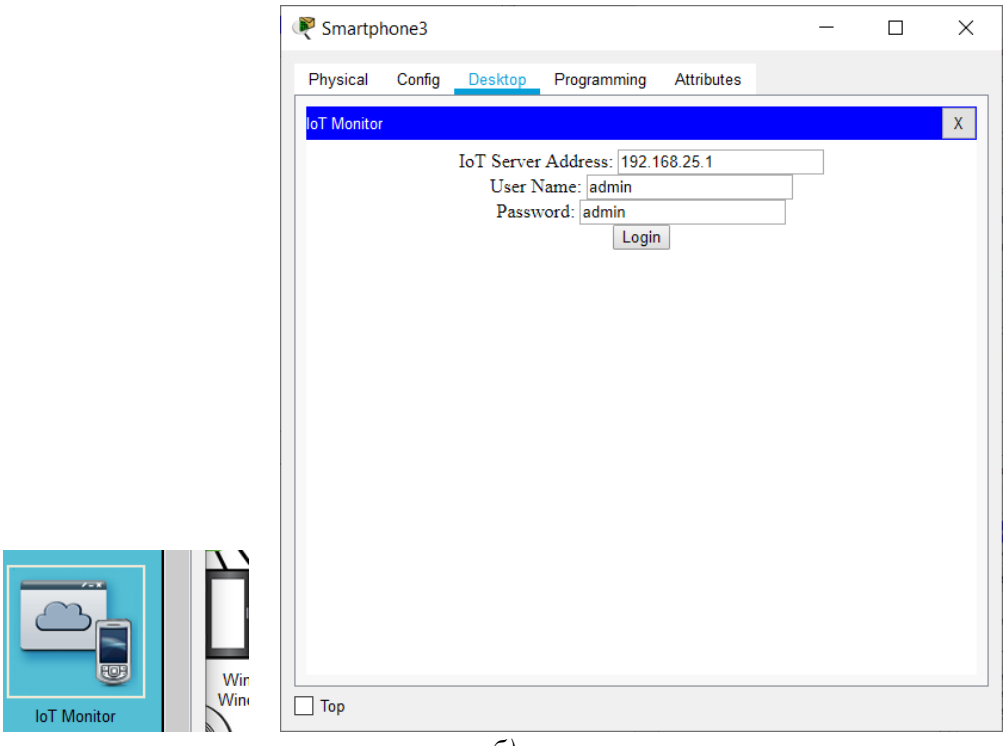
❖ Осъществяване на дотъп до IoT устройствата на системата за пожароизвестяване и пожарогасене през Smart phone устройството посредством въвеждане на адреса на мрежата в обхвата на Home Gateway фиг. 13.18. Процедурата може да бъде направена чрез стандартния Web Browser (фиг. 13.19.а)) или приложението IoT Monitor (фиг. 13.19.б)). Независимо от избрания вариант е необходимо въвеждането на съответно потребителско име и парола за достъп;



Фиг. 13.18. *Home Gateway LAN мрежови адрес*
за достъп и управление на системните устройства



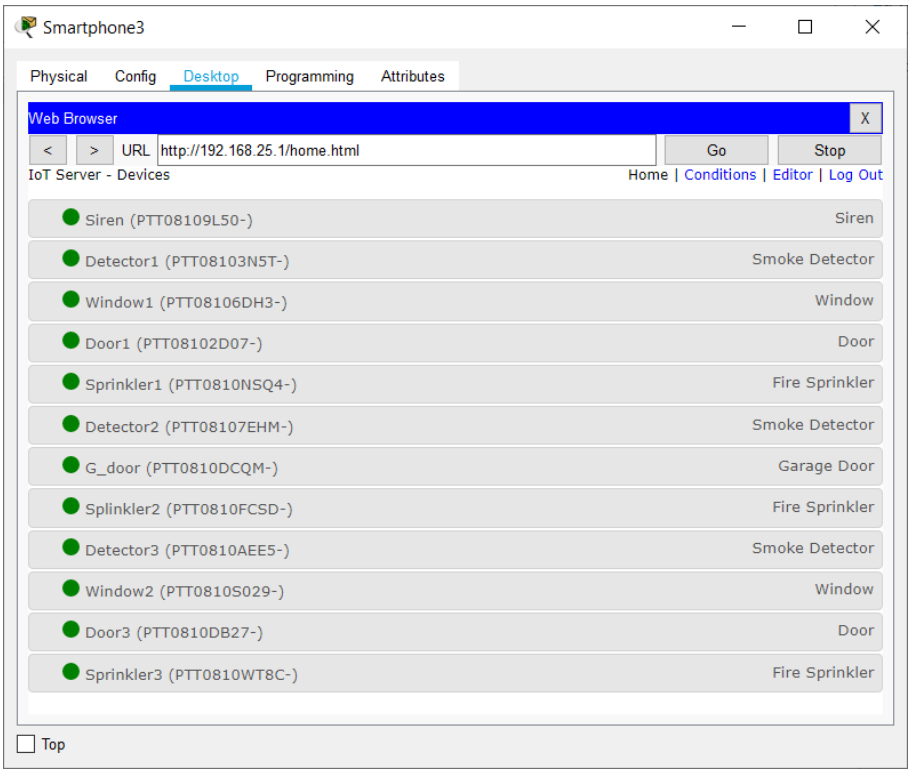
а)



б)

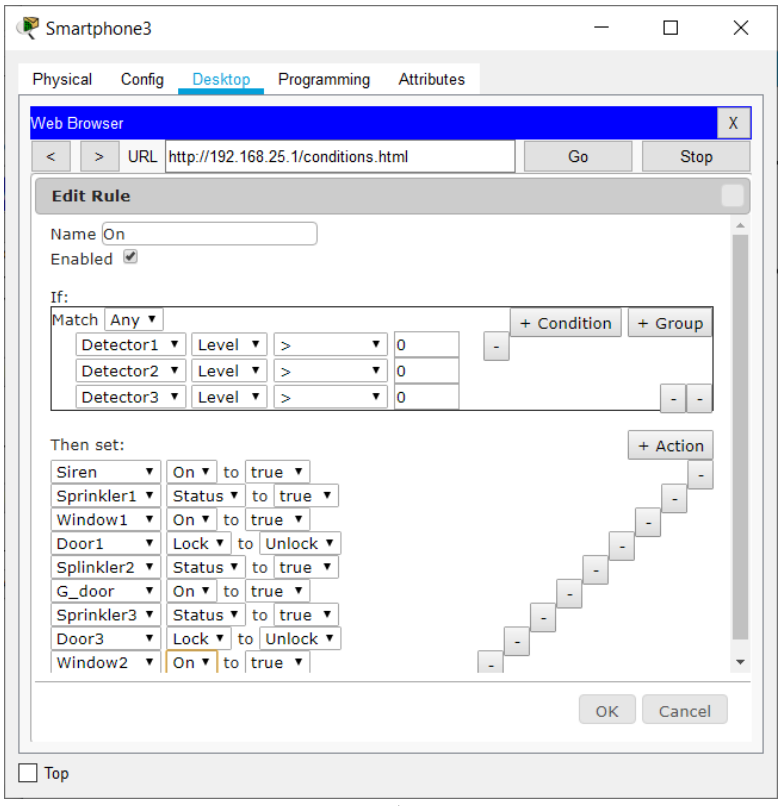
Фиг. 13.19. *Достъп до системните IoT устройства през а) Web Browser и б) IoT Monitor приложение*

При коректна оторизация от страна на потребителя наличните IoT устройствата ще бъдат видими и достъпни за последваща конфигурация и управление, както е представено на фиг. 13.20.

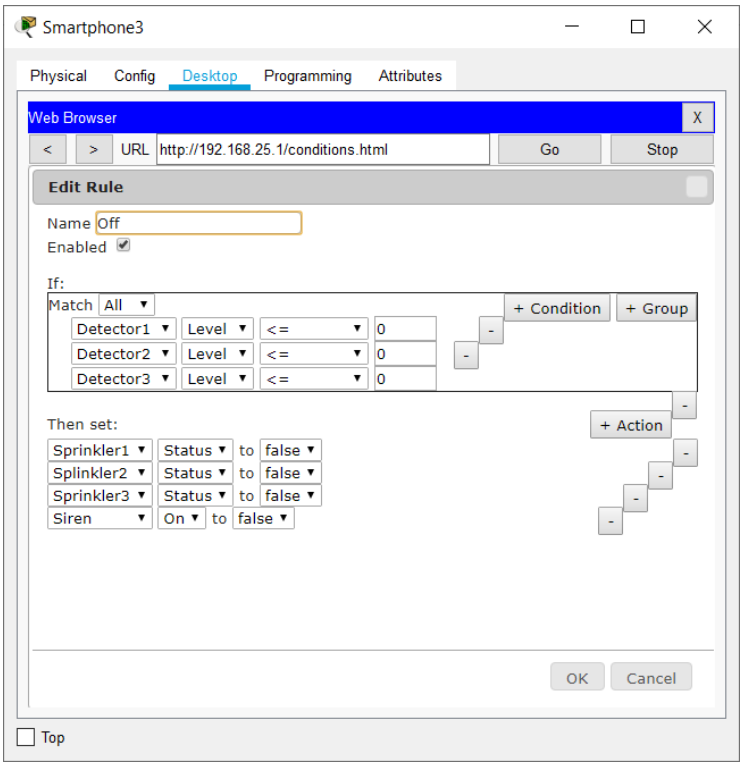


Фиг. 13.20. *IoT устройства за конфигурация и управление*

❖ Особено важна стъпка след установяване на видимост и достъп до IoT системните устройства е дефинирането на ***два типа условия*** относно тяхната активност при 1) детектиране на дим/възникване пожар или 2) отсъствие на дим/възникнали пламъци:



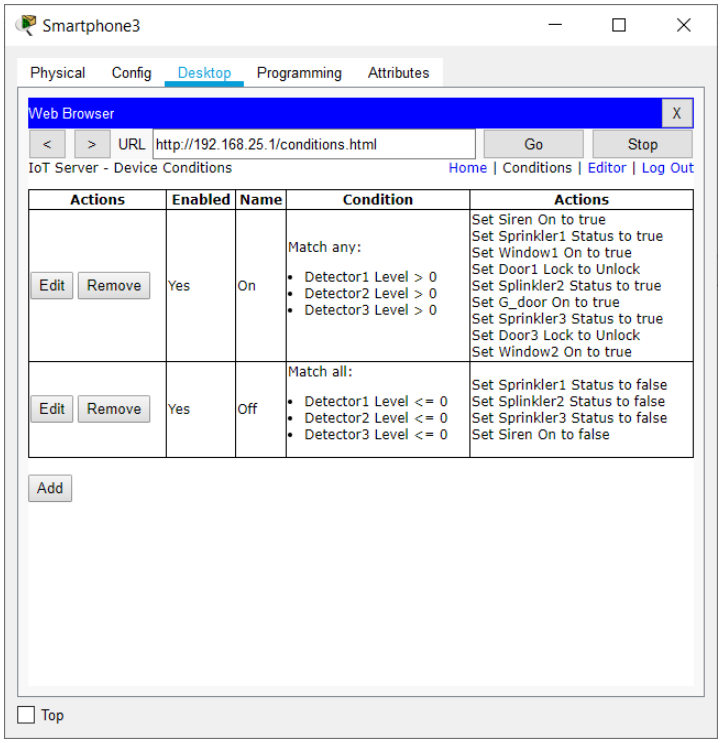
а)



б)

Фиг. 13.21. *Условия а) „On“ и б) „Off“ относно активация на IoT устройствата*

- Първият тип правила регментират задействане на механизмите за изпускане на вода, звукова сигнализация, отключване на вратите в обособените Kitchen и Bedroom, отваряне на гаражната врата и прозорците при наличие на активация на детекторите за дим – Условия „***On***“, дадени на фиг. 13.21.а);

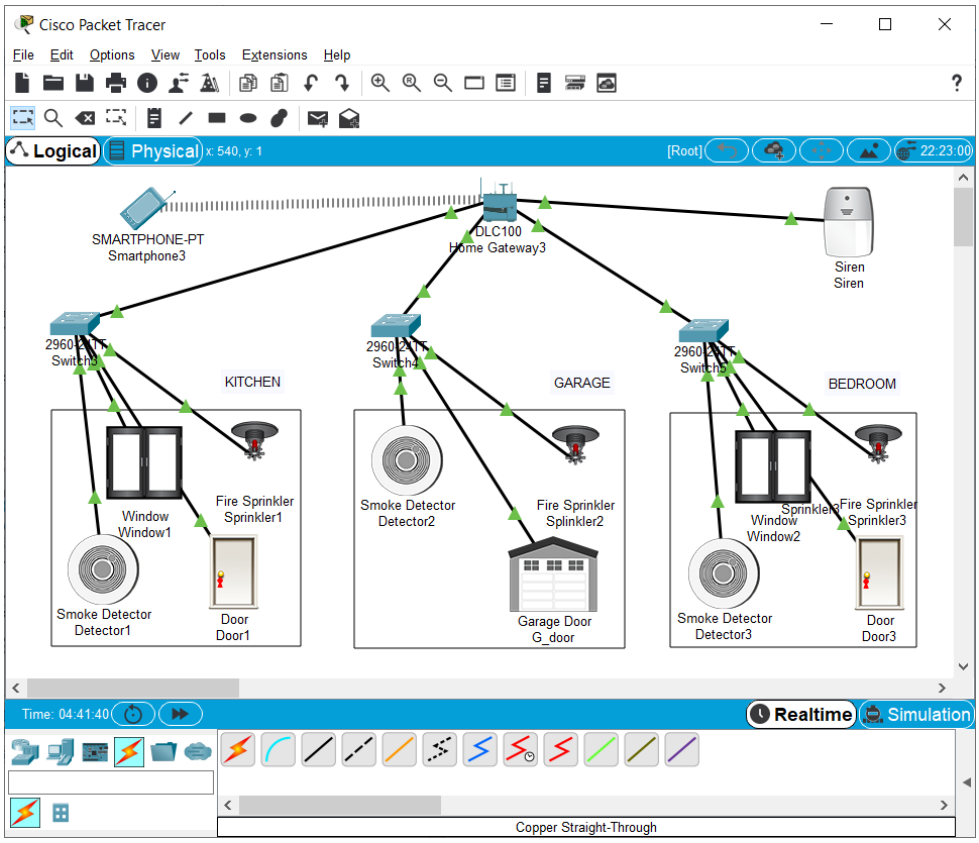


Фиг. 13.22. *Краен вид на Условия „On“ и „Off“ за IoT управление*

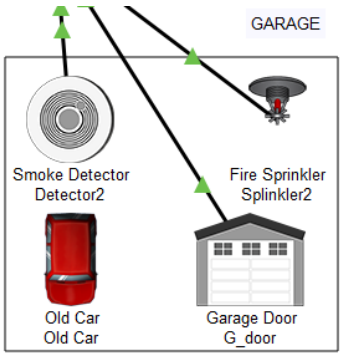
- Вторият тип условия засяга изключване на устройството за звуково пожароизвестяване и спиране на пожарогасителните пръскачки като прозорците и гаражната врата остават отворени, а вратите на помещения Kitchen и Bedroom отключени - Условия „***Off***“, посочени на фиг. 13.21.б).

Дефинираните правила за актизация на системните устройства са представени на фиг. 13.22.

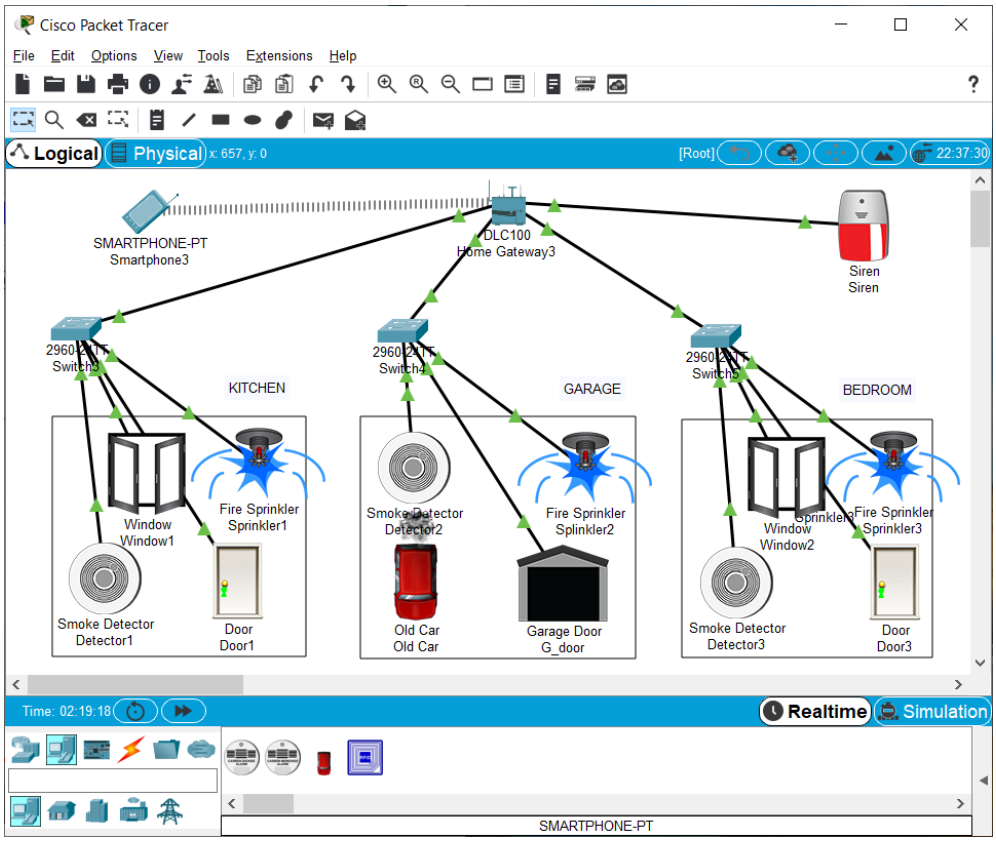
❖ Последният процедурен етап се състои в тестване на работоспособността на ситемата за пожароизвестяване и пожарогасене на основата на IoT технология, чиято топология е показана на фиг. 13.23. Във връзка с тествовия процес е необходим източник на дим, който да задейства някой от детекторите. В случая може да бъде използвана „стара кора“, разположена в гаражното помещение на фиг. 13.24 (IoT устройство Old Car се задейства чрез комбинация “*alt+press*” като показалеца на мишката се намира върху него).



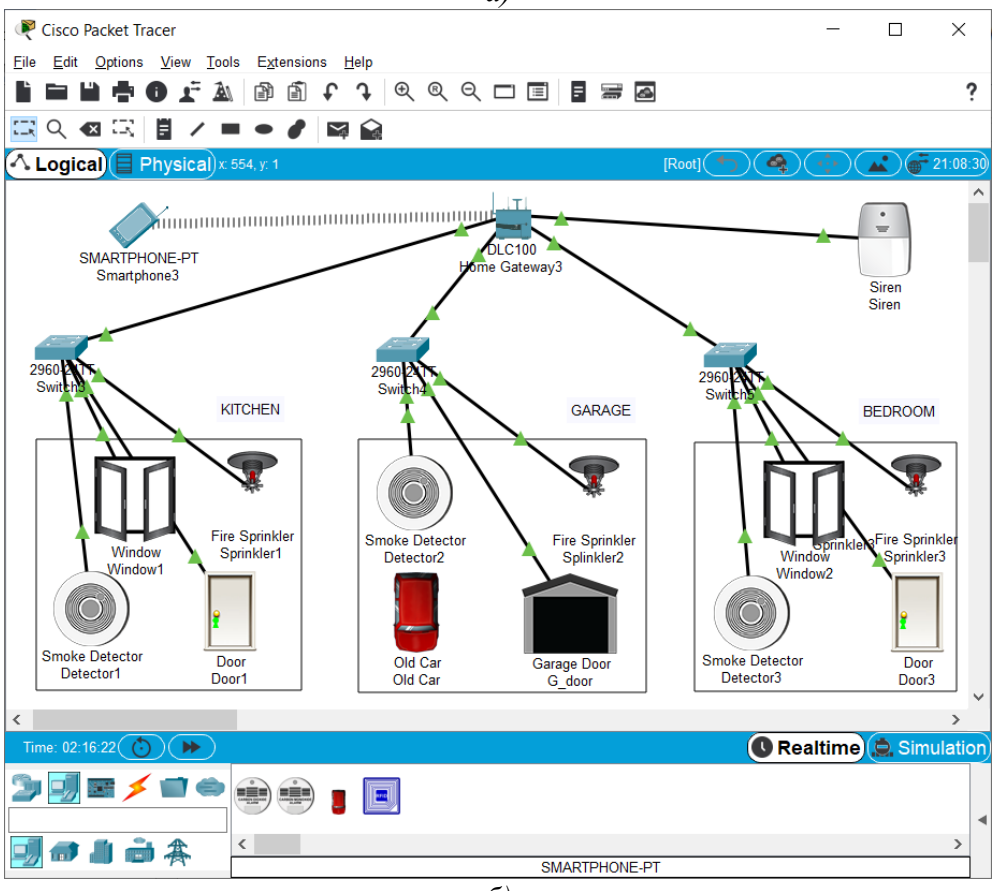
Фиг. 13.23. *Модел на системата за пожароизвестяване и пожарогасене при тестване*



Фиг. 13.24. *IoT устройство “Old Car*



а)



б)

Фиг. 13.25. *Изгледи при тестване на системата при а) констатирано наличие на дим/пожар и б) потушаване на пожара*

За на тестване на системата са имитирани два сценария, съответно „наличие на дим/възникване на пожар“ и „отсъствие на дим/потушаване на пожара“, илюстрирани на фиг. 13.25.

13.4. Контролни въпроси.

1. Какви категории технически средства се използват за удостоверяване на детектирането на дим / огън в системите за пожароизвестяване?
2. Кои типове детектори според контролирания параметър се прилагат в системите за пожароизвестяване?
3. Какви видове детектори в пожароизвестителните системи се включват според конфигурацията?
4. Какви са основните типове пожароизвестителни системи, изграждани в административни, бизнес и индустриални обекти?
5. Посредством коя процедура IoT устройствата стават „видими“ или достъпни за управление чрез WEB Browser или приложението IoI Monitor от персонализираните потребителски станции за мониторинг?
6. В какво се състои същността на дефинираните “on” и “off” групи от условия за управление на сензорните елементи и изпълнителните механизми в моделираната IoT система за пожароизвестяване и пожарогасене?

Лабораторно упражнение № 14

IOT УПРАВЛЕНИЕ НА КОНЦЕПТУАЛНИ СИСТЕМИ „ИНТЕЛИГЕНТЕН ДОМ“ В СИМУЛАЦИОННА СРЕДА CISCO PACKET TRACER

14.1. Цел на упражнението

Да се запознаят студентите с възможностите за контрол на оторизацията на потребителкият достъп при управление на Smart Home приложения с IoT технология с помощта на конфигурирани Radius & Registration Servers.

14.2. Теоретична постановка

Интернет на нещата (IoT) представлява глобална инфраструктура за информационното общество, позволяваща съвременни услуги от взаимосвързани (физически и виртуални) източници със съдържание на оперативно съвместима информация да бъдат достъпни чрез иновативни комуникация технологии. Фигура 14.1 показва основните характеристики, следващи от технологията.

Фиг. 14.1. *Характеристики на IoT технологията*

Технологиите за активиране на Интернет на нещата могат да бъдат групирани в три основни категории:

- технологии, които позволяват на „нещата“ да събират и натрупват контекстуална информация;
- технологии, даващи възможност на "нещата" да извършват обработка на контекстуални данни;
- технологии за подобряване на сигурността и поверителността.

Първите две категории могат да бъдат реализирани самостоятелно или съвместно като функционални градивни елементи, изискващи изграждане на понятието „интелигентност“. Третата категория не се характеризира с функционалност, а се определя по-скоро като изискване. На фиг. 14.2 са представени технологиите за реализация на IoT [27].

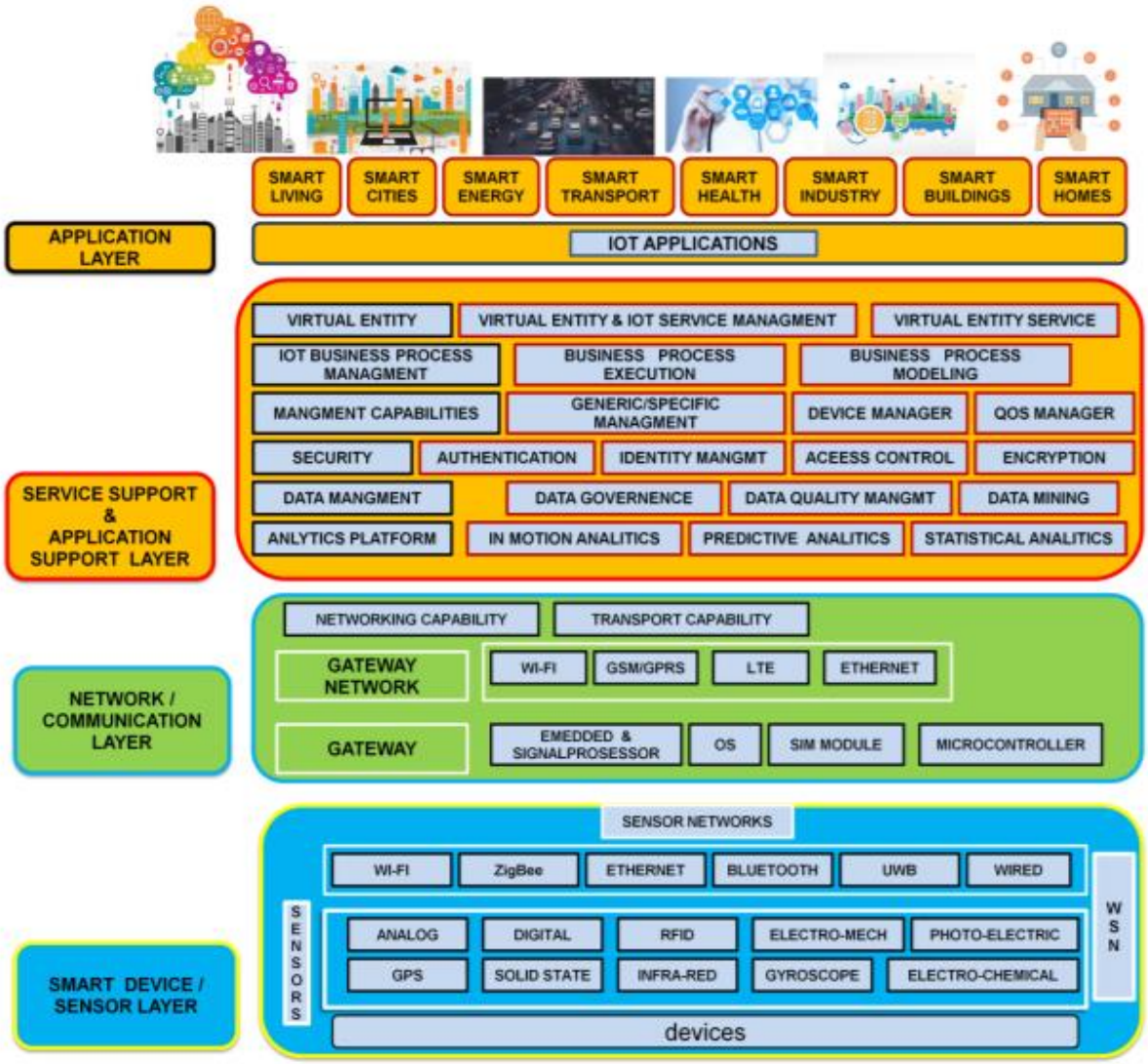
Фиг. 14.2. *Технологии за реализация на IoT*

Някои от по-значимите са:

- Сензорни мрежи;
- Облачни изчисления;
- Вградени системи и системна интеграция;
- Информационна и сигнална обработка;
- Нанотехнологии и др.

IoT архитектура и бъдещи технологии

IoT архитектурата се състои от различни слоеве на технологии, поддържащи IoT – фиг. 14.3. Технологиите са свързани една с друга и комуникират на основата на различни сценарии по отношение на мащабируемост, модулност и конфигурация. Основните функционални слоеве са, респективно:



Фиг. 14.3. *IoT архитектура*

- Интелигентни устройства / сензорен слой – сензорните елементи позволяват свързаност с околната среда и мониторинг на основни нейни параметри (температура, влажност, скорост и посока на движение на въздука, налягане, проводимост и т.н.). Появява се изискване за свързване на сензорите към Sensor Gateways, които могат да се бъдат под формата на локална мрежа (LAN) като Ethernet и Wi-Fi връзки или лична мрежа (PAN), като ZigBee, Bluetooth и Ultra Wideband (UWB). Особено популярни са безжичните сензорни мрежи WSN;
- Gateways и Networks – сензорните елементи генерират огромен обем от данни, който изисква надеждна и високоскоростна кабелна или безжична инфраструктура за информационно предаване. Това налага използването на ново поколение частни, публични или хибридни мрежови модели, изградени на базата на определени комуникационни изисквания относно латентност,

честотна лента и сигурност – изисква се внедряването на хетерогенна конфигурация;

- Управление на услугите и ресурсите – тук се включват действия по аналитична обработка на сензорна и друга информация, контрол на сигурността, моделиране на различни интерактивни и автоматизирани прицеси, управление на IoT устройствата, обекти и системи, инструменти за реакция спрямо непредвидени събития, филтриране на данните и т.н.;
- Приложен слой – тук се засягат различни сфери като транспорт, строителство, градска среда, селско стопанство, здравеопазване, туризъм и кулрура, енергетика и околна среда [27, 28].

Развитието на редица технологии в това число полупроводниковата електроника, комуникациите, сензорите, смарт телефоните, вградените системи, облачните мрежи, виртуализацията и други определят същественото значиние на софтуерните инструменти, позволяващи на физическите устройства да се адаптират и функционират безпрепятствено в променяща се среда и да бъдат свързани през цялото време навсякъде. В тази връзка в таблица 14.1 са обобщени бъдещите разработки и изследователски нужди, обвързани с концепцията „Интернет на нещата“ [27, 28].

Таблица 14.1. Бъдещи разработки и изследователски нужди относно технологии, свързани с концепцията „Интернет на нещата“

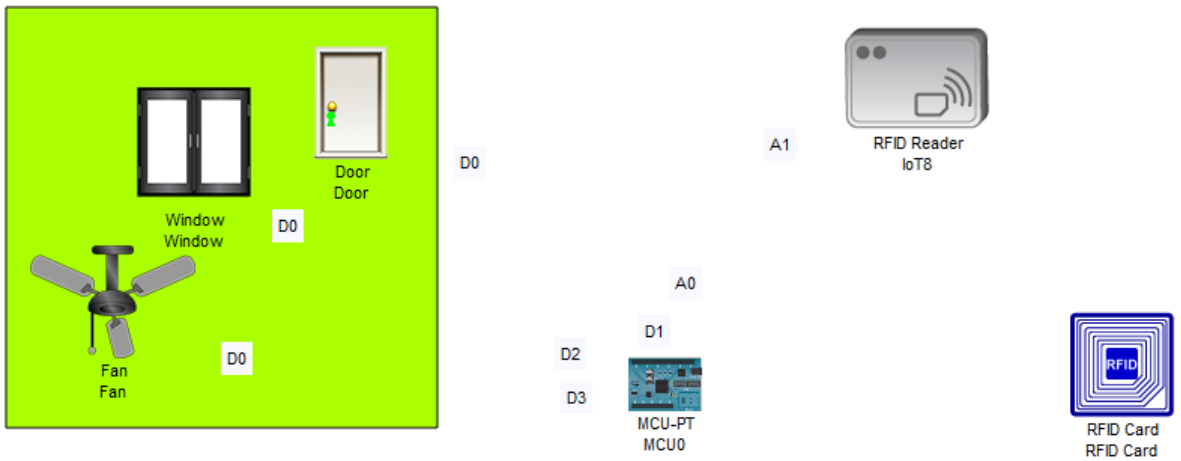
Технология	**Бъдещи разработки**	**Изследователски нужди**
Communication Technology	•On chip antennas •Wide spectrum and spectrum aware protocols •Unified protocol over wide Spectrum •Multi-functional reconfigurable chips	•Protocols for interoperability •Multi-protocol chips •Gateway convergence •On chip networks •Longer range (higher frequencies – tenths of GHz) •5G developments
Network Technology	•Self aware and self organizing networks •Self-learning, self-repairing networks •IPv6- enabled scalability •Ubiquitous IPv6-based IoT deployment	•Grid/Cloud network •Software defined networks •Service based network •Need based network
Data and Signal Processing Technology	•Context aware data processing and data responses •Cognitive processing and optimization •IoT complex data analysis •IoT intelligent data visualization •Energy, frequency spectrum aware data processing	•Common sensor ontology •Distributed energy efficient data processing •Autonomous computing
Discovery and Search Engine Technologies	•Automatic route tagging and identification management centers •On demand service discovery/integration	•Scalable Discovery services for connecting things with services
Security & Privacy Technologies	•User centric context-aware privacy and privacy policies •Privacy aware data processing •Security and privacy profiles selection based on security and privacy need	•Low cost, secure and high performance identification/ authentication devices •Decentralized approaches to privacy by information localization

14.3. Последователност на работа и задачи за изпълнение

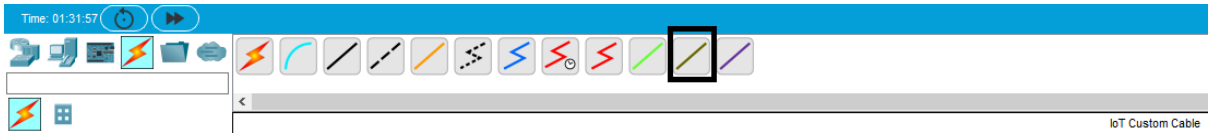
14.3.1. IoT – управление на потребителския достъп чрез RFID четец и карта

С оглед на концепцията „Интелигентен дом“ и интегриране на IoT технологията една от основните задачи се отнася до персонализация при контрола на достъпа с помощта на съвременни заключващи системи за дома. Една от възможностите за подобен вид системи се свързва с използването на RFID четци и карти за контрол на достъпа.

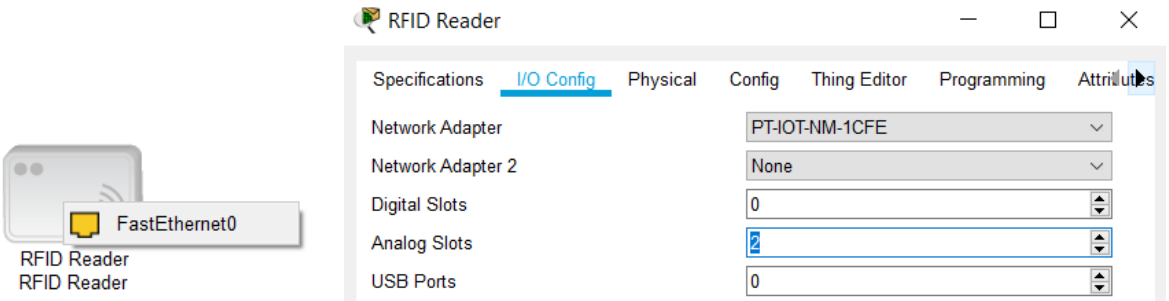
На фиг. 14.4 е представен сценарии на RFID Активиране / Блокиране на потребителския достъп с включване на “RFID Reader” “RFID Card” до IoT – базирани устройства „Входна врата - Door“, „Прозорец - Window“ и „Таванен вентилатор - Fan“ в обособено имитирано помещение. Конфигурацията на оказаните устройства е реализирано на основата на модул MCU-PT, който разполага с набор от цифрови и аналогови Входове / Изходи. Връзката между устройствата и портовете на многофункционалната платка се осъществяват с “IoT Custom Cable” – фиг. 14.5.



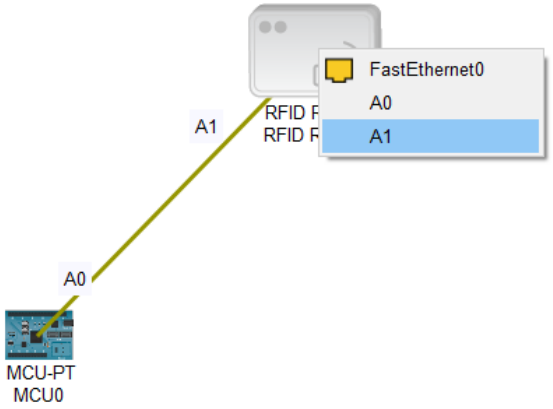
Фиг. 14.4. *IoT контрол на достъпа до имитирано помещение при концепция „Интелигентен дом“ чрез RFID активация*



Фиг. 14.5. *Достъп до “IoT Custom Cable” в лентата с типове връзки между мрежови устройства “Connections”*



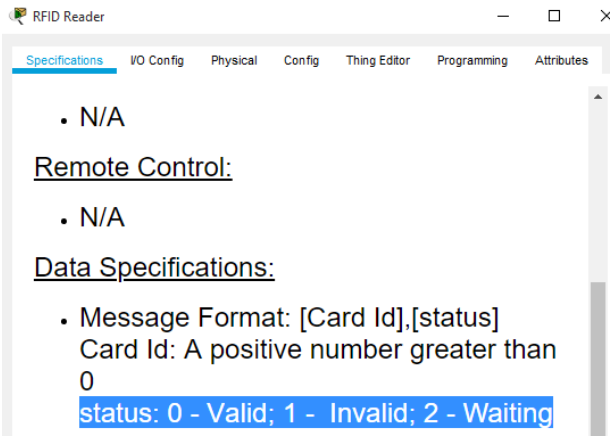
а) *б)*



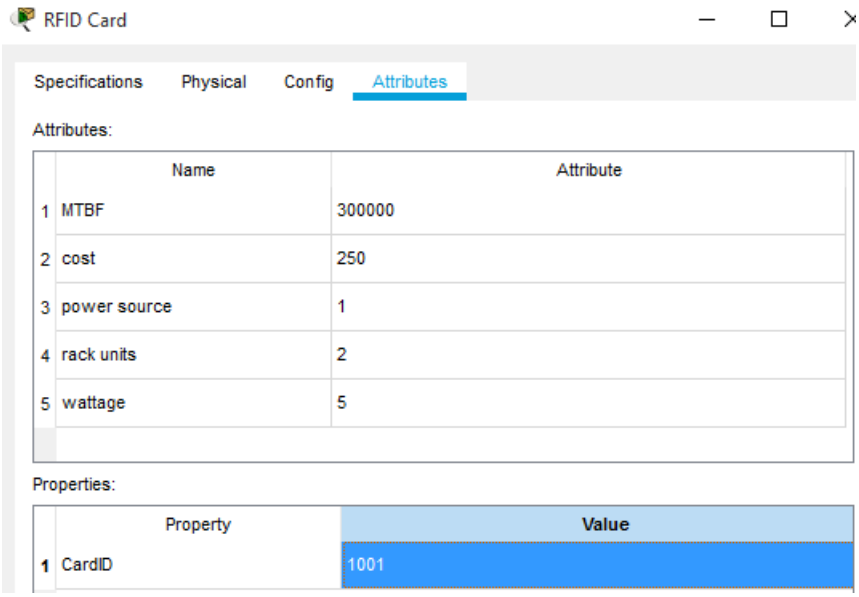
в)

Фиг. 14.6. *Добавяне на Analog Slots за връзка към модул MCU-PT*

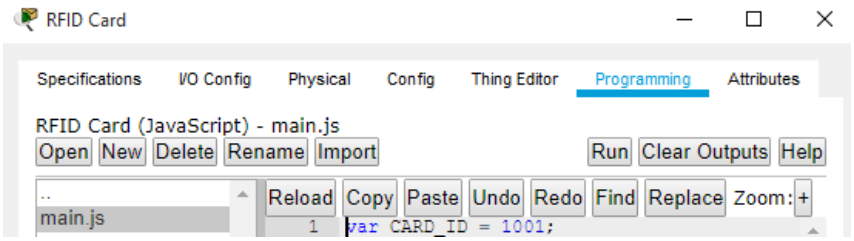
Тъй като RFID четецът разполага само с Fast Ethernet интерфейс – фиг. 14.6.а), за да бъде реализирана връзка към модул MCU-PT, е необходимо да бъдат добавени аналогови слотове при съответната опция, достъпна чрез избор на меню I/O Config, – фиг. 14.6.б), след което вече те ще бъдат налични на фиг. 14.6.в).



Фиг. 14.7. *Статус на RFID картата, поддържани от RFID Reader*



а)

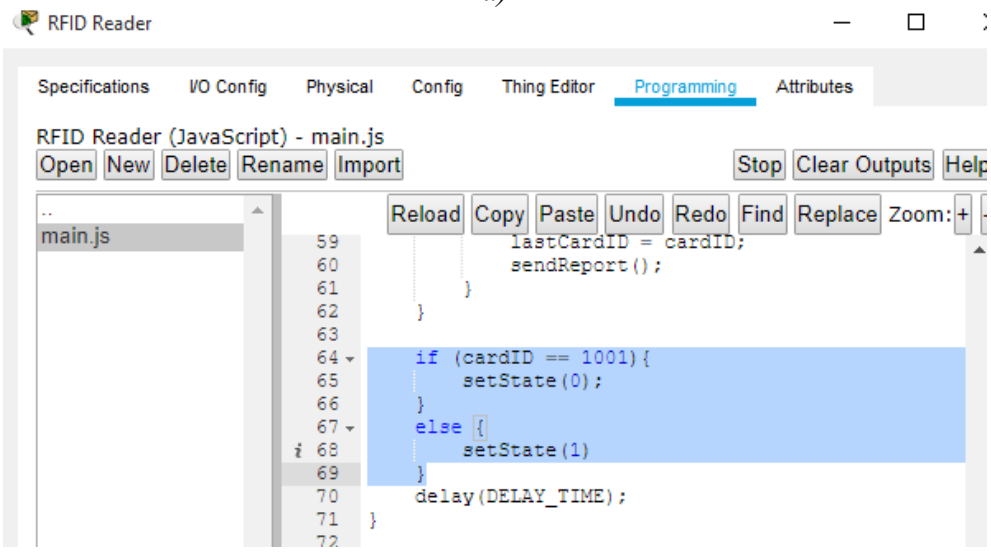


б)

Фиг. 14.8. *RFID Card идентификатор – а) меню "Attributes" и б) JavaScript шаблон*

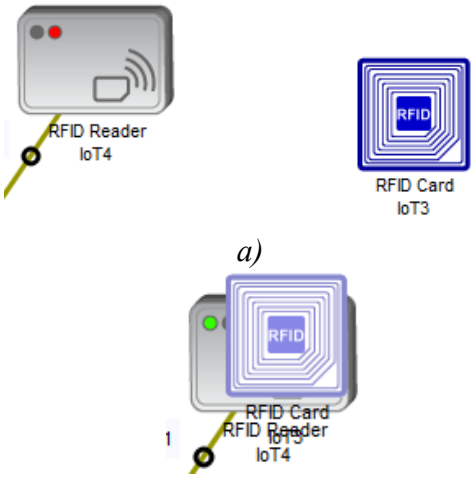
За идентификация на състоянието на RFID четеца са налични три опции, поддържани от него "валидна – status 0", „невалидна - status 1" или оторизация „в режим на изчакване - status 2", както се вижда на фиг. 14.7. Асоциирането на конкретна карта със съответен четец се базира на конкретния идентификатор Card ID, с който тя разполага (фиг. 14.8). Съответният RFID Card идентификатор следва да бъде заложен в програмния JavaScript шаблон на четеца и обвързан с посочените по-горе състояния на картата. Процедурата се реализира чрез добавяне на маркирания сорс код в програмната функция „Loop" (фиг. 14.9.а), показано на фиг. 14.9.б). Активирането на четеца вече е възможно като се извършва с доближаване на RFID картата в непосредствена близост до неговия обхват, както е показано на фиг. 14.10.

а)

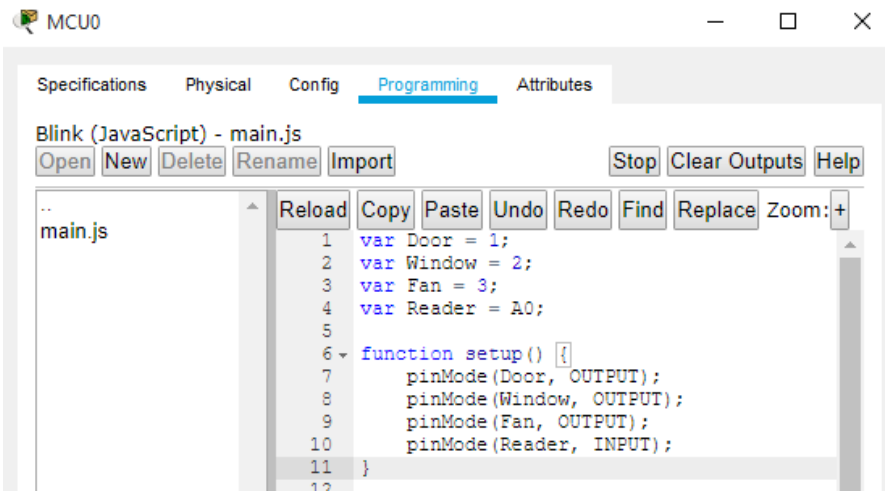


б)

Фиг. 14.9. *Достъп до "function loop" а) и задаване на RFID Card идентификатор и асоцииране на състояния "валидна – status 0", „невалидна - status 1" б)*



Фиг. 14.10. *Активиране а) и деактивиране б) на RFID четеца чрез доближаване на и отдалечаване на картата*

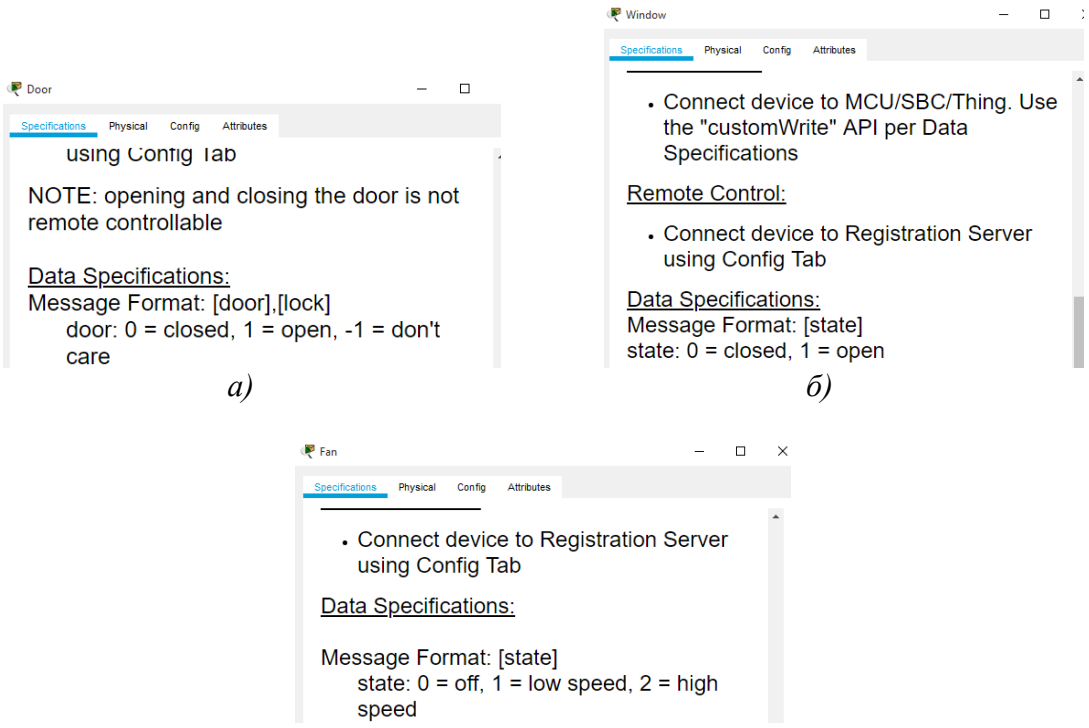


Фиг. 14.11. *Присвояване на цифрови изходи D1, D2 и D3 и аналогов вход A0 за контрол на достъпа до устройствата в имитираното помещение и следене на потребителската оторизация чрез RFID Reader*

Следващата фаза от конфигурацията, свързана с платка MCU-PT, се състои в настройка на цифровите портове D1, D2 и D3 като изходи за управление на активацията на потребителските IoT устройства “Door”, “Window” и “Fan”, както и назначаване на аналоговия порт A0 в състояние „вход“ относно следене на състоянието на RFID четеца. За целта се въвежда програменият сегмент в JavaScript темплейта от фиг. 14.11. Поредна стъпка е задаването на процедурно IoT активиране / деактивиране – „function loop“, съответно:

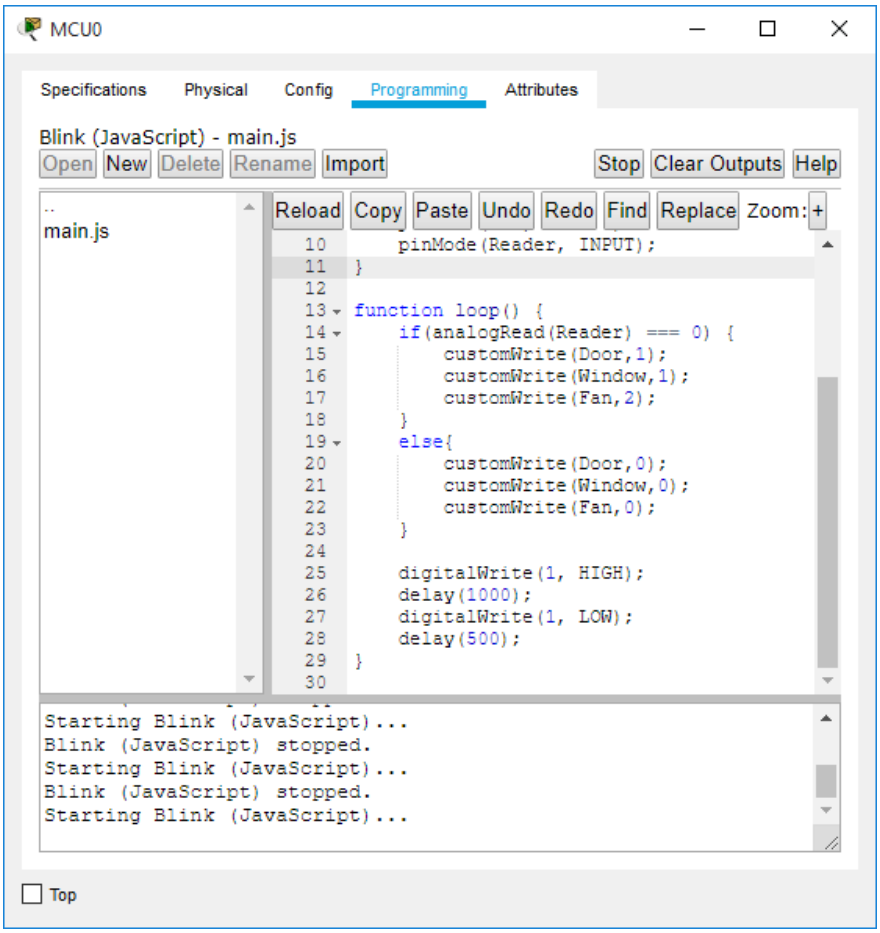
- „отваряне“ на „входната врата“ и „прозореца“ – (преглед на статуса по дефиниция на фиг. 14.12.а) и фиг. 14.12.б));
- „включване на висока степен на въртене“ на „вентилатора“ - (преглед на статуса по дефиниция на фиг. 14.12.в)) – *използван за демонстрация на възможност за активиране на IoT устройства с различно предназначение от включваните в стандартните заключващи системи „врати“ и „прозорци“*,

при наличие на коректна оторизация за персонализирани потребители (фиг. 14.15).

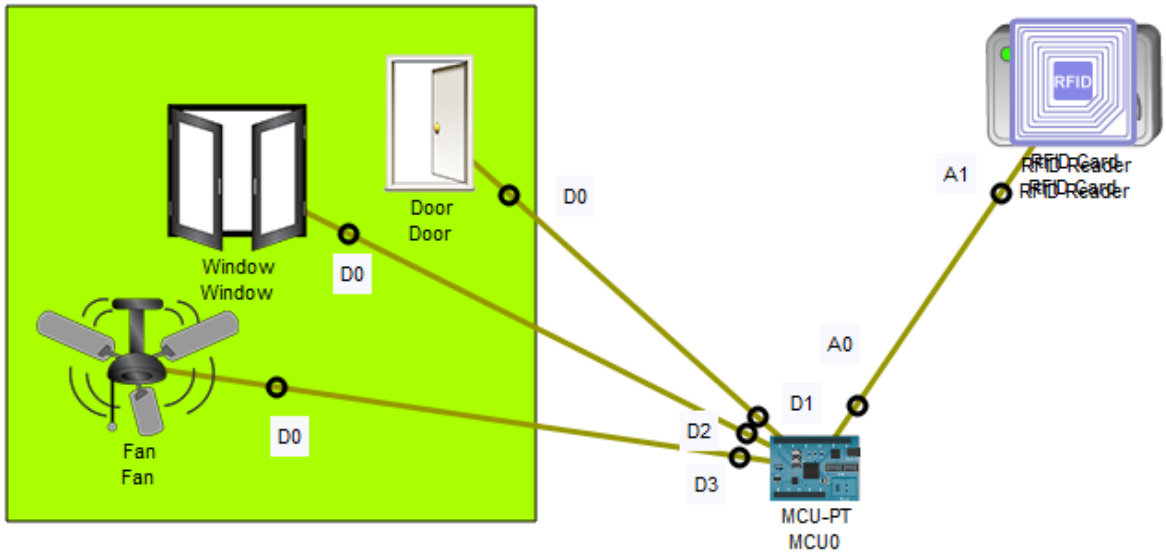


а) *б)* *в)*

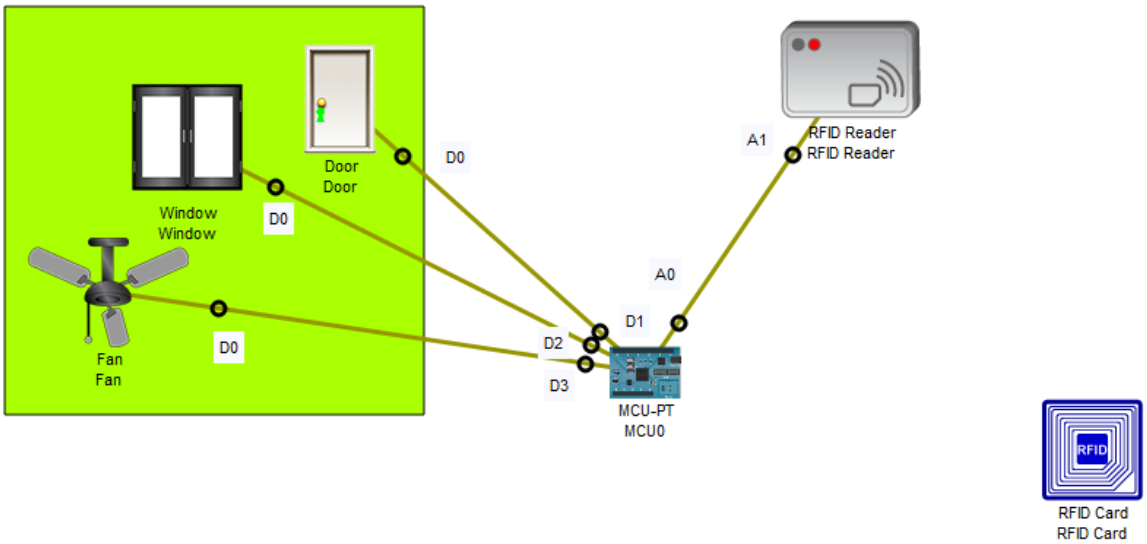
Фиг. 14.12. *Преглед на възможните състояния на а) IoT Door, б) IoT Window и в) IoT Fan*



Фиг. 14.15. *Активиране / Деактивиране на IoT Door, Window и Fan устройства при коректен потребителски достъп*



а)



б)

Фиг. 14.16. *Тестване на мрежовата конфигурация при констатирана коректна верификация на потребителска RFID карта – а) Активиране и б) Деактивиране на IoT Door, Window и Fan устройства в имитирано помещение*

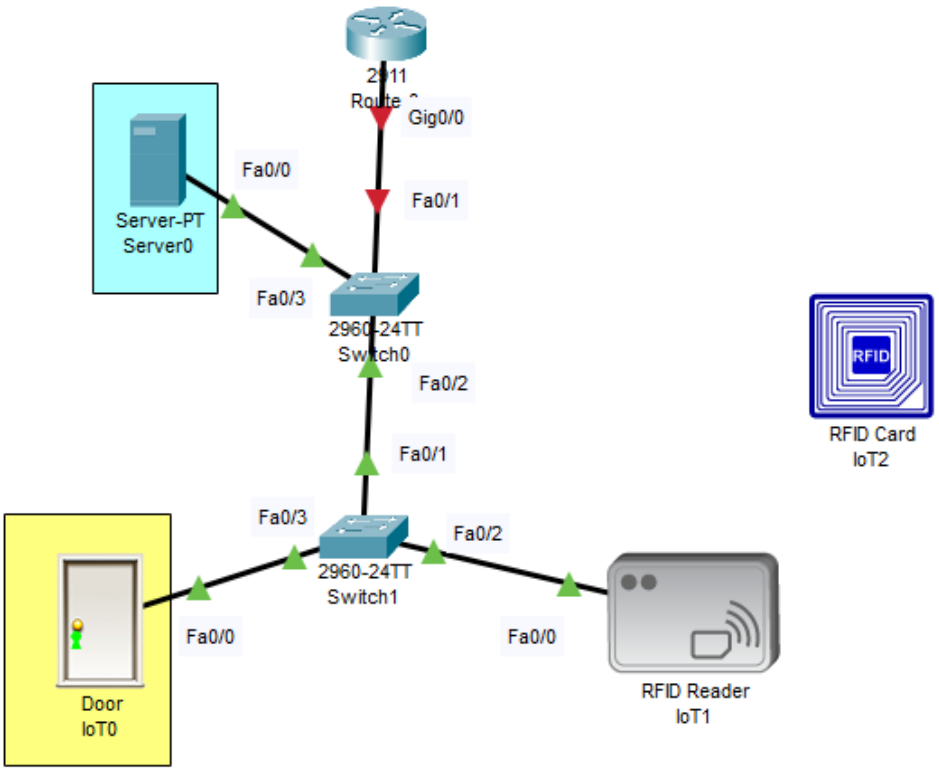
Фигура 14.16 представя последната фаза от проведеното процедурно тестване с верификация на контрола на достъп за персонализирани потребители, притежаващи RFID карта с унифициран идентификатор “1001”. Както може да бъде видяно, активацията на устройствата с IoT управление е успешна. Разглежданият сценарии е базисен и позволява модифициране с добавяне на:

- ❖ нови устройства;
- ❖ допълнителни мрежови сегменти;
- ❖ нови RFID карти,

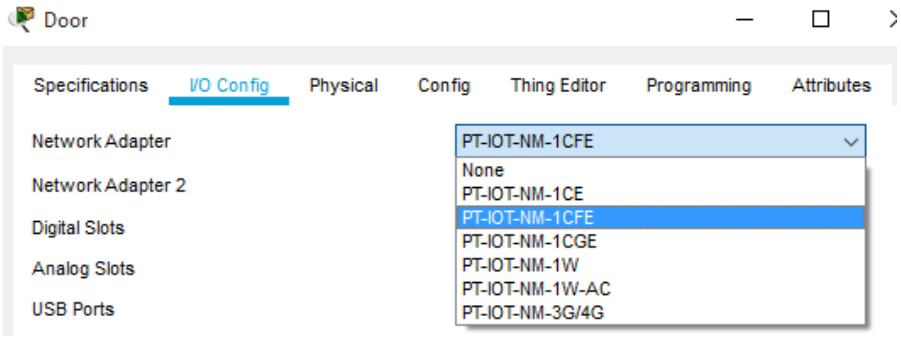
както и адаптиране на други технологични инструменти и решения за мониторинг на достъпа.

14.3.2. Моделиране на RFID заключваща система с мониторинг на достъпа чрез Registration Server

Подобен сценарии с контрол на достъпа чрез RFID четец и карта позволява включването на сървърно звено, изпълняващо ролята на Registration Server, достъпен през създадени потребителски профили за мониторинг и управление на IoT устройства в мрежа с динамична IP адресация и свързаност между мрежовите устройства с медна преносна среда. Примерната конфигурация, включваща мрежови устройства “Server-PT”, маршрутизатор “Router 2911”, комутатори “Switch 2960-24TT”, входна врата с RFID – базиран достъп и управление “IoT Door”, “RFID Reader” и “RFID Card”, е представена на фиг. 14.17. За да бъде осъществена връзка с „входната врата“ е необходимо да бъде използван Fast Ethernet интерфейс чрез избор на специализиран мрежови адаптер тип “PT-IOT-NM-1CFE”, както е показано на фиг. 14.18.



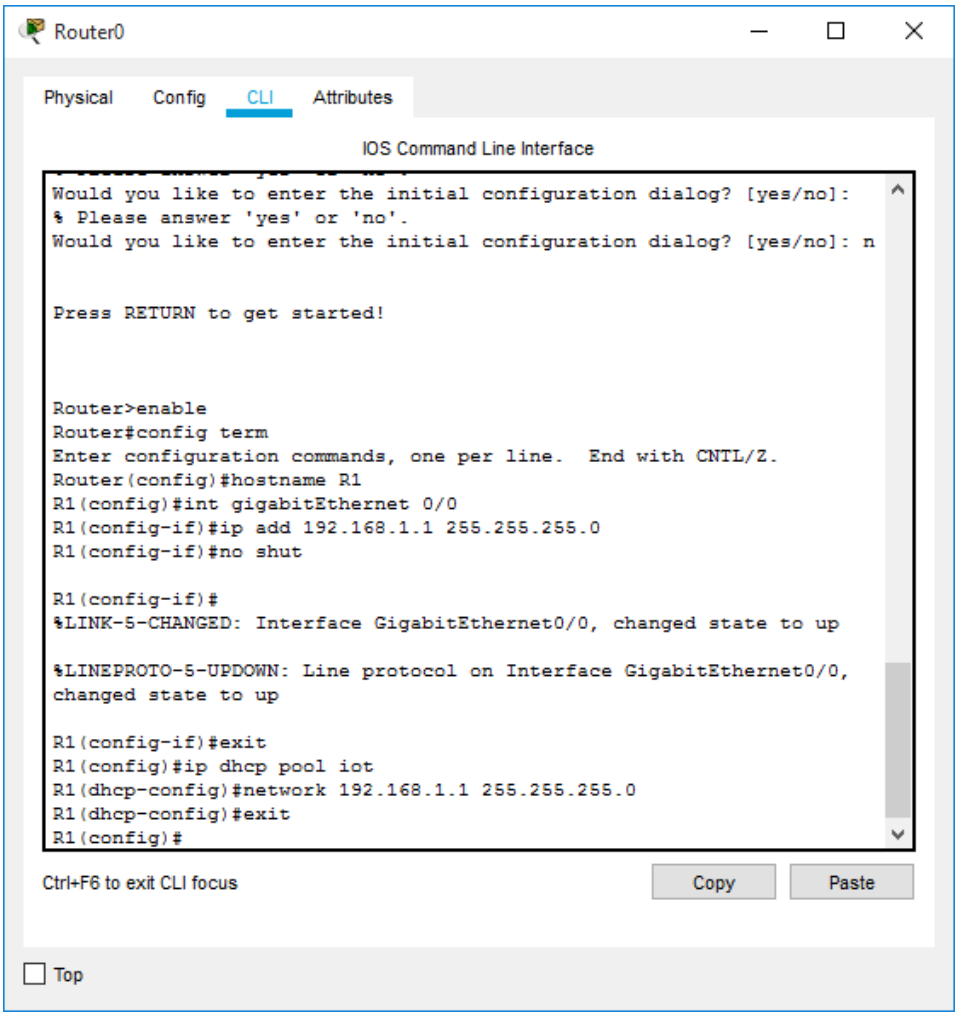
Фиг. 14.17. *Мрежовата конфигурация на RFID заключваща система и Registration Server управление на потребителския достъп през медна преносна среда*



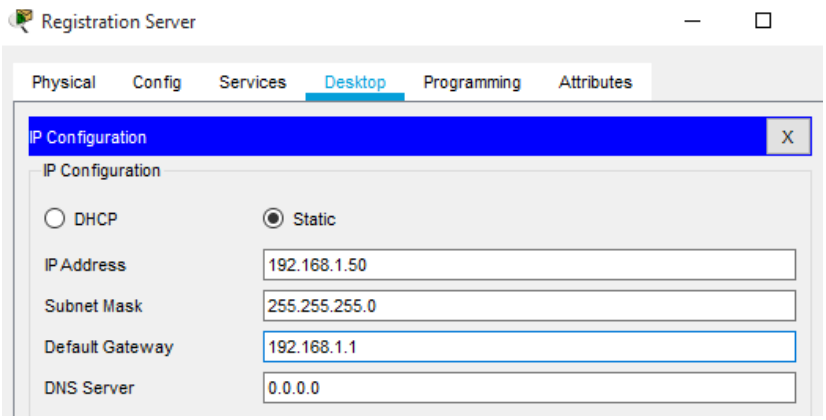
Фиг. 14.18. *Избор на мрежови адаптер относно тип интерфейс Fast Ethernet*

Първата стъпка от след свързване на мрежовите устройства се отнася до IP и DHCP конфигурация на интерфейса GigabitEthernet 0/0 и назначаване на обхват за

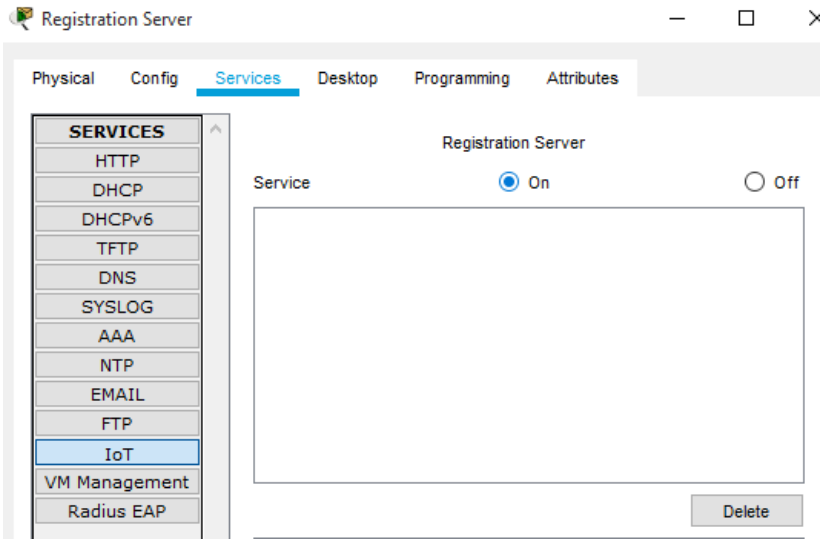
потребителска мрежа “iot” за динамично назначаване на IP адреси в Command Line Interface на маршрутизатора с хост име “R1”. Кодът е даден на фиг. 14.19. След това се преминава към IP конфигуриране и активиране на услугата “Registration Server” със структурното сървърно звено, показано на фиг. 14.20. Създава се потребителски профил за мониторинг на обекта за достъп от моделираната заключваща система „IoT Door“ и четеца “RFID Reader” чрез въвеждане на адреса на сървъра в Web браузъра и следване на последователността, представена на фиг. 14.21.



Фиг. 14.19. *Настройка на режима за динамично назначаване на IP адреси в обхвата на потребителската мрежа “iot”*

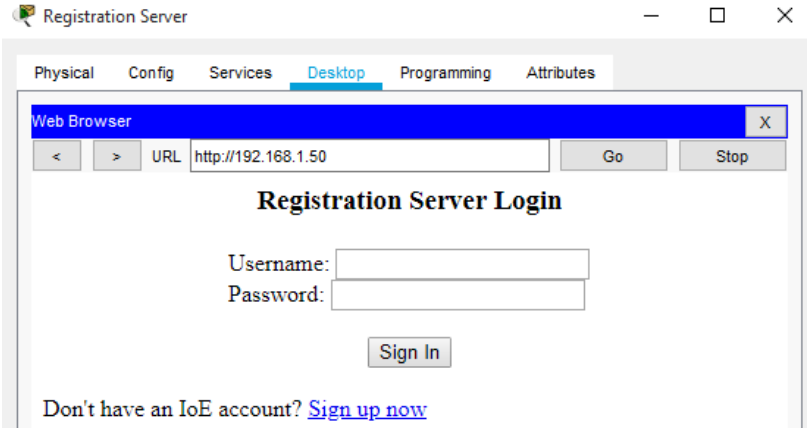


а)

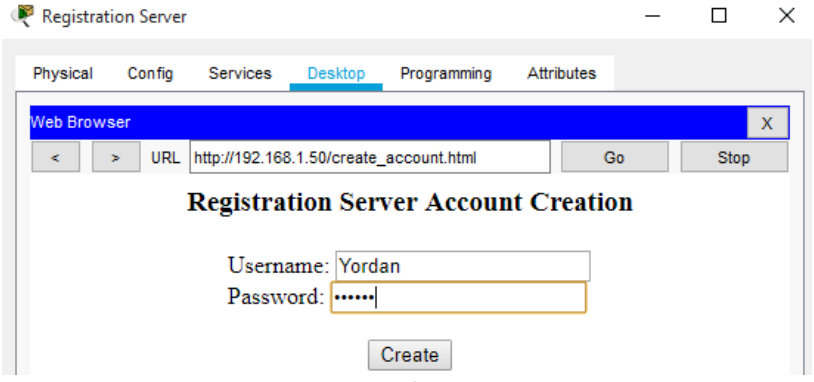


б)

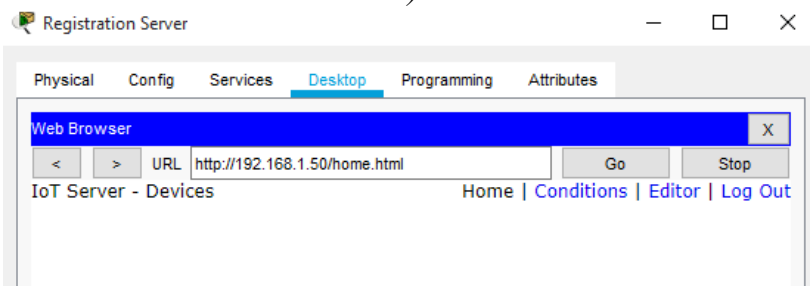
Фиг. 14.20. *IP конфигурация а) и активация на услуга "Registration Server" б)*



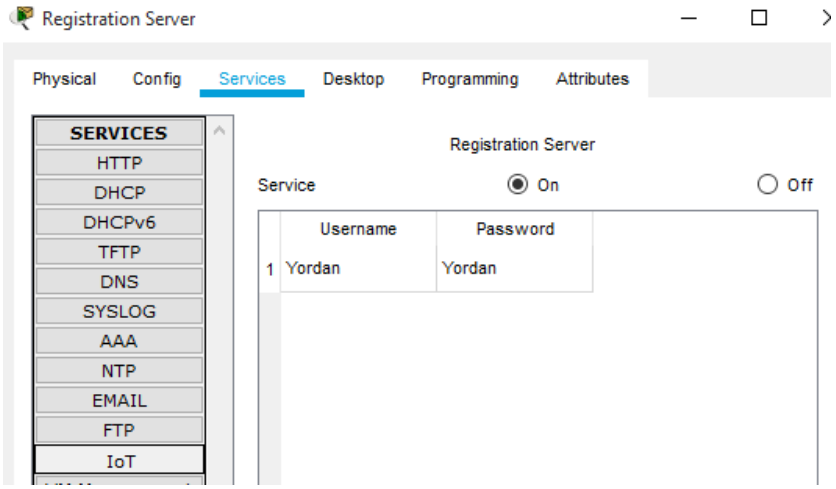
а)



б)

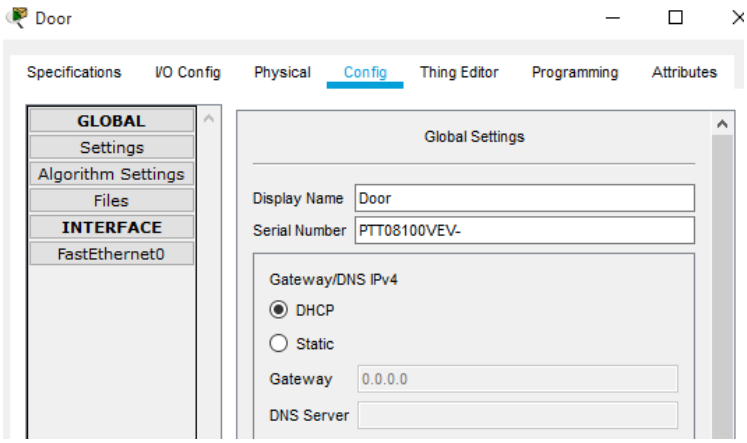


в)

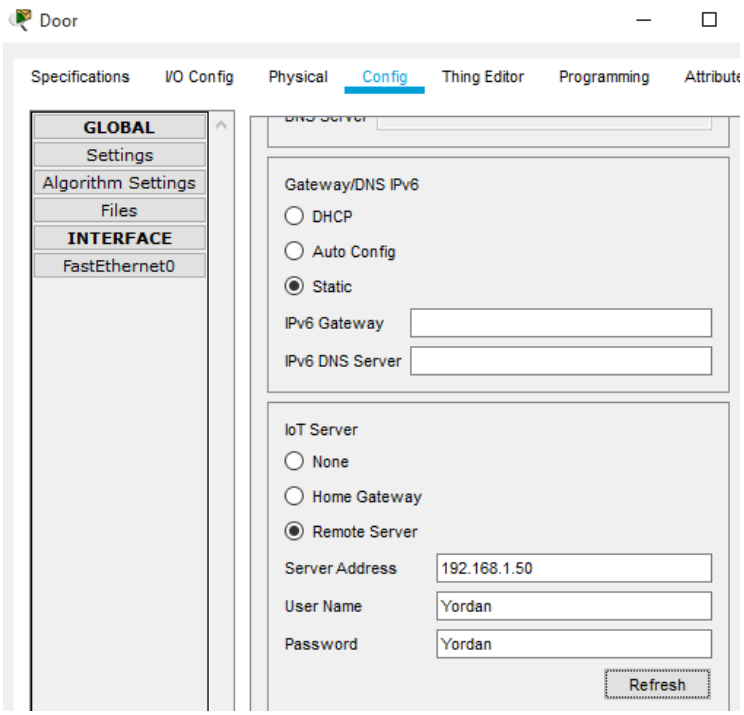


г)

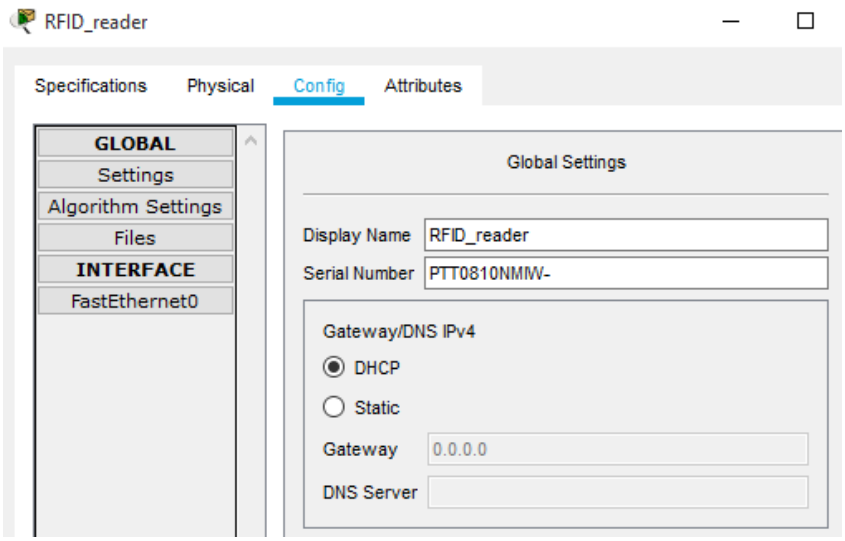
Фиг. 14.21. *Създаване на потребителски профил към "Registration Server" за мониторинг на избрани IoT устройства за следене и управление*



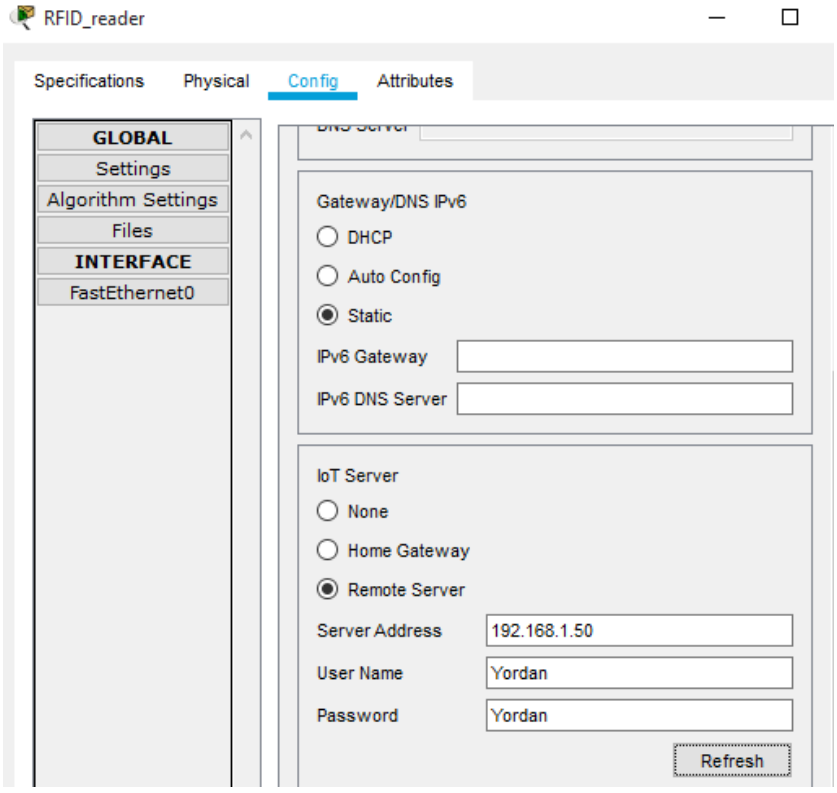
а)



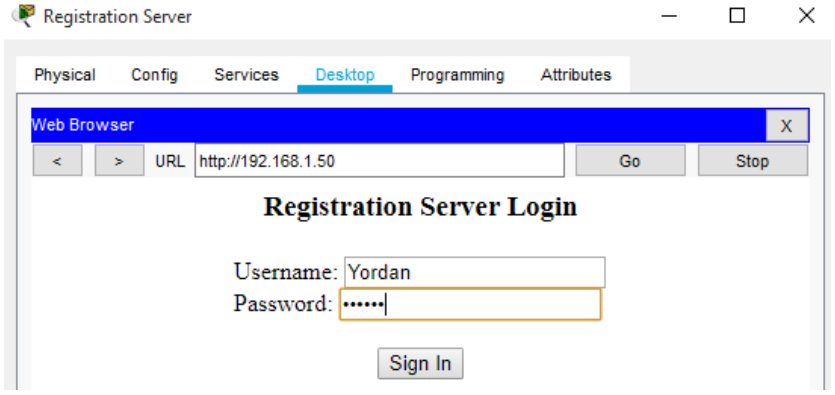
б)


RFID_reader
Specifications
Physical
Config
Attributes
GLOBAL
Settings
Algorithm Settings
Files
INTERFACE
FastEthernet0
Global Settings
Display Name
RFID_reader
Serial Number
PTT0810NMIW-
Gateway/DNS IPv4
DHCP
Static
Gateway
0.0.0.0
DNS Server

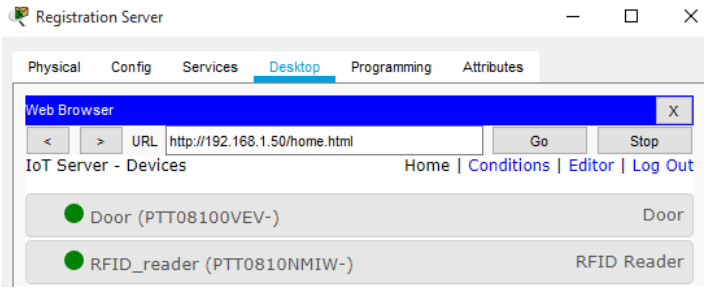
в)


RFID_reader
Specifications
Physical
Config
Attributes
GLOBAL
Settings
Algorithm Settings
Files
INTERFACE
FastEthernet0
Gateway/DNS IPv6
DHCP
Auto Config
Static
IPv6 Gateway
IPv6 DNS Server
IoT Server
None
Home Gateway
Remote Server
Server Address
192.168.1.50
User Name
Yordan
Password
Yordan
Refresh

г)


Registration Server
Physical
Config
Services
Desktop
Programming
Attributes
Web Browser
URL
http://192.168.1.50
Go
Stop
Registration Server Login
Username:
Yordan
Password:
Sign In

д)

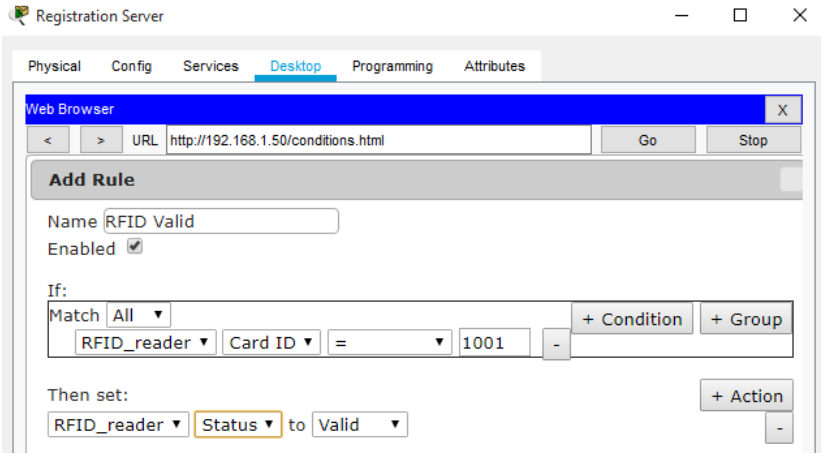


е)

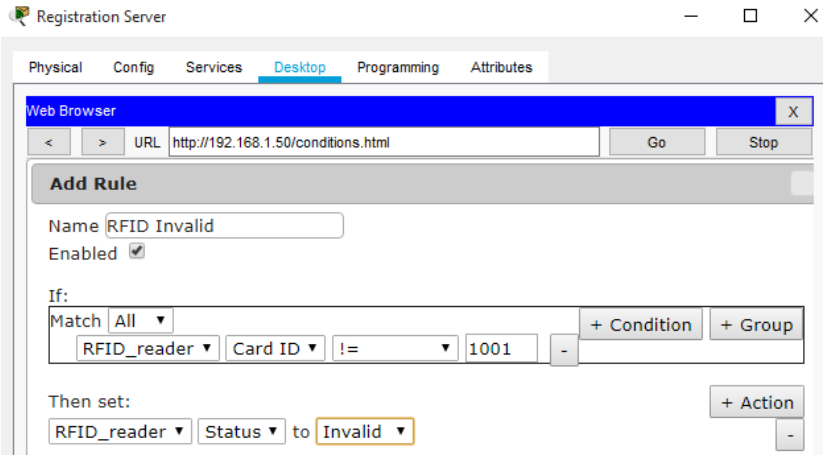
Фиг. 14.22. *Създаване на връзка към "Registration Server" (а) и б) за „входната врата"; в) и г) при RFID четеца) и проверка на видимостта на устройствата за следене и контрол (д) и е))*

След това е необходимо да бъде:

- осъществена връзка между "IoT Door", "IoT RFID Reader" и мониторинговия "Registration Server" – процедура за динамично назначаване на шлюз, тип IoT Server "Remote Server" за конкретен потребителски профил – фиг. 14.22.а) до фиг. 14.22.г);
- направена верификация на коректността за достъп до целевите мрежови устройства чрез оторизация на указания профил - – фиг. 14.22.д) и фиг. 14.22.е).



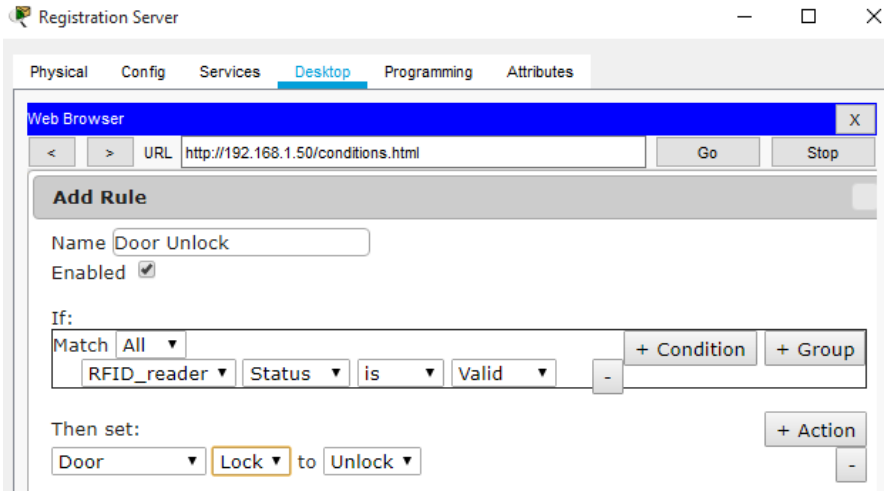
а)



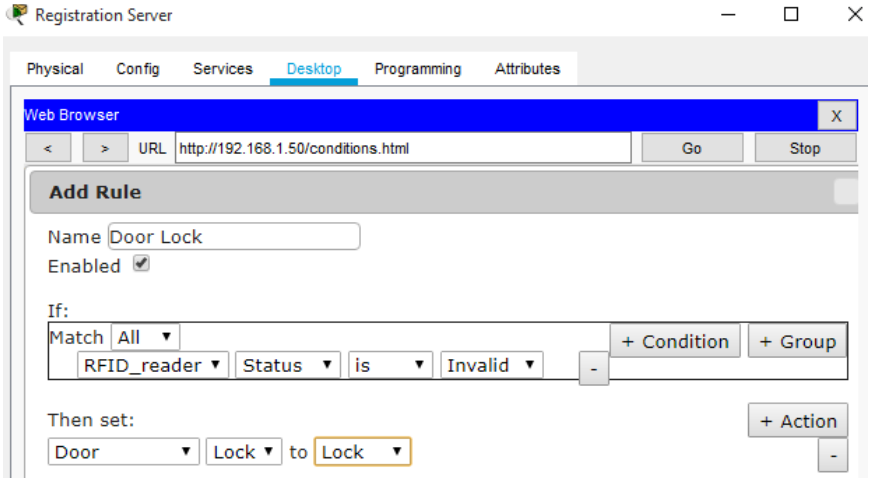
б)

Фиг. 14.23. *Задаване на а) „валиден" и б) „невалиден" достъп с асоциирана на точен RFID Card идентификатор*

На следващ етап се провеждат действия по активиране на RFID четеца спрямо асоцииран идентификатор на RFID картата за достъп. Посочените дейности, свързани със указване на условия “Conditions” за „валидна“ (коректна RFID карта) или „невалидна“ RFID карта с различна принадлежност от тази на потребителя) оторизация, са онагледени на фиг. 14.23.

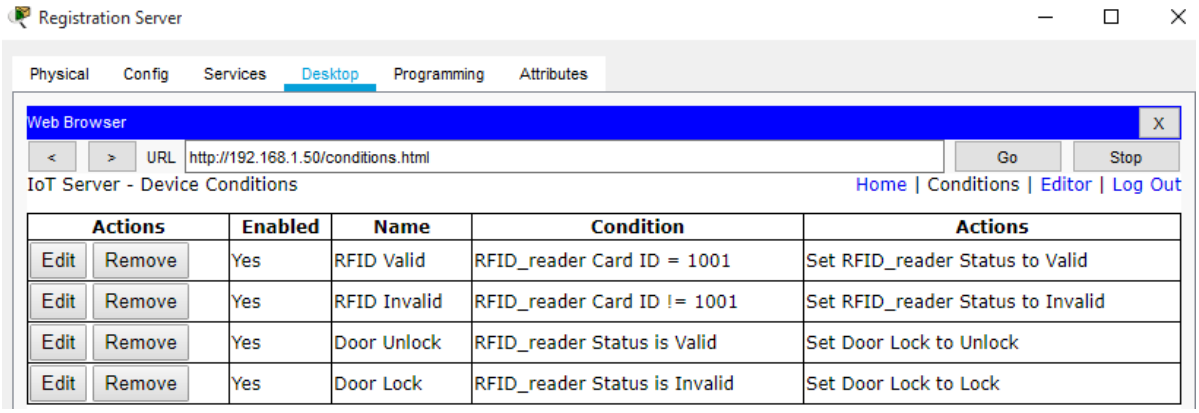


а)



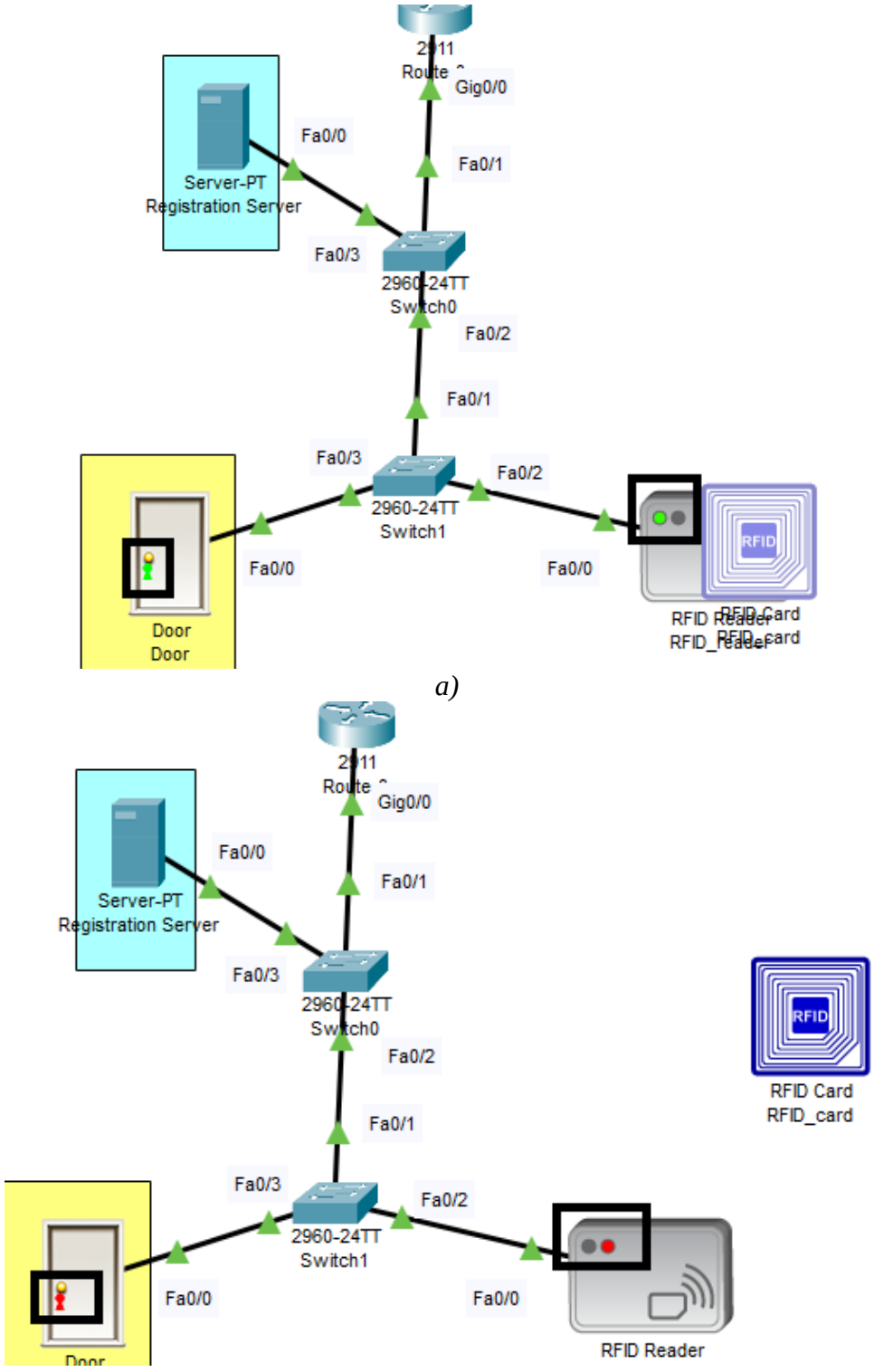
б)

Фиг. 14.24. *Присвояване на състояние „Unlock – отключване“ при валиден а) и б) „Lock – заключване“ при установена невалидност на статуса на RFID четеца*



Actions	Enabled	Name	Condition	Actions
Edit Remove	Yes	RFID Valid	RFID_reader Card ID = 1001	Set RFID_reader Status to Valid
Edit Remove	Yes	RFID Invalid	RFID_reader Card ID != 1001	Set RFID_reader Status to Invalid
Edit Remove	Yes	Door Unlock	RFID_reader Status is Valid	Set Door Lock to Unlock
Edit Remove	Yes	Door Lock	RFID_reader Status is Invalid	Set Door Lock to Lock

Фиг. 14.25. *Общ вид на условията за контрол на достъпа при мрежовата заключваща система*



а)

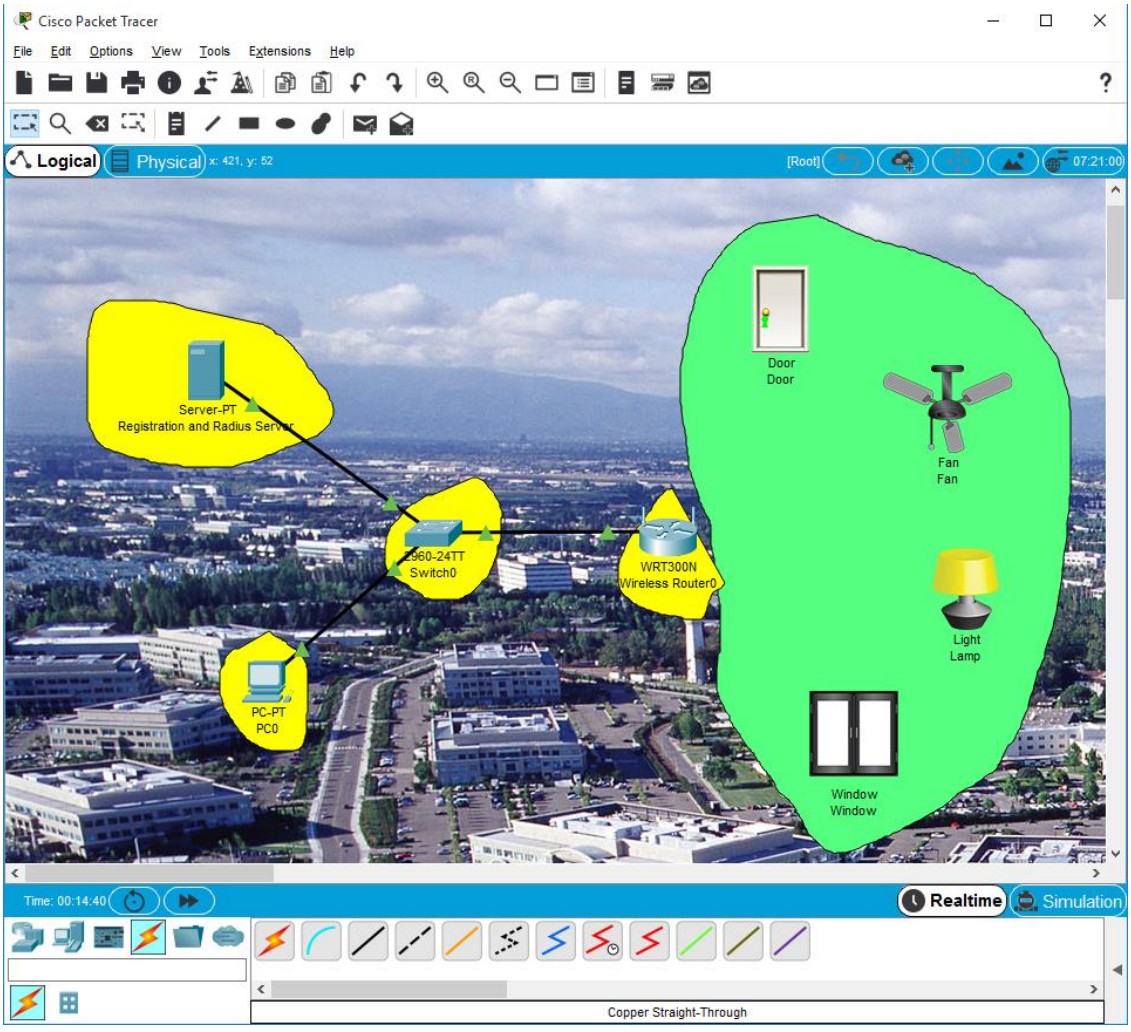
б)

Фиг. 14.26. *Тестване на активността на достъп при RFID заключващата система*

По аналогичен начин на фиг. 14.24 се задават процедури по „отключване“ и „заключване или оставяне в заключено положение“ на “IoT Door” при установена коректна / некоректна оторизация на създаден потребителски профил. Общ изглед на четирите създадени условия за мониторинг на достъпа през звено “Registration Server” е представен на фиг. 14.25, а на фиг. 14.26 е направено тестване на работоспособността на моделираната мрежова конфигурация на RFID заключваща система.

14.3.3. Моделиране на система „Интелигентен дом“ за мониторинг и управление на IoT устройства на базата Radius и Registration Server

Разглежда се задачата за изграждане на мрежа за достъп на потребители до IoT устройства по безжична връзка при дефиниран режим на сигурността и криптографски подход на основата на конфигурация на Radius (Remote Authentication Dial-In User Server) сървър и мониторинг на устройствата през Web чрез Registration сървър. На фиг. 14.27 е представена структура на мрежовите устройства, участващи при проектиране на концепция за „Интелигентен дом“. Фигурата онагледява частично свързване на персонална компютърна станция за достъп до и управление на IoT устройствата PC0, комутатор 2960-24TT, безжичен рутер WRT300N и Registration and Radius сървърно звено Server-PT.



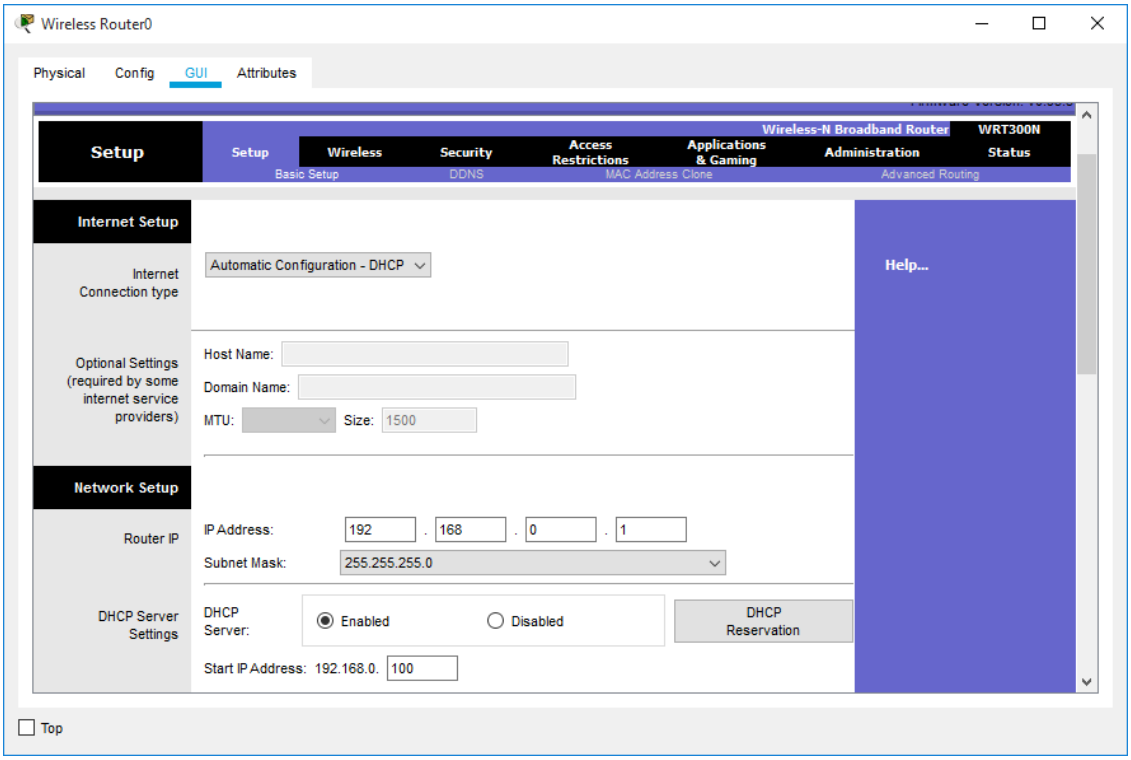
Фиг. 14.27. *Частично свързване на устройства от система “Интелигентен дом” на основата на Registration and Radius Server*

Първата стъпка конфигурацията се състои в задаване на серия от настройки за дефиниране на Radius сървър и назначаване на идентификационни имена и пароли на всяко от IoT устройствата, които ще бъдат обект на отдалечен мониторинг и контрол. Radius сървърът представлява механизъм за подобряване на Wi-fi мрежовата сигурност като генерира WAP2 Enterprise протокол за безжична точка за достъп. WAP2 е протокол за сигурност, използващ IEEE 802.1X стандарт. Конкретната персонална потребителска станция PC0 реализира достъп до IoT устройствата през осигурена WAP2 защитена

връзка между Radius сървъра и безжичния рутер WRT300N при използване на AES (Advanced Encryption Standard) метод на криптиране. Тук се извършват дейности по „автентикация“ (Authentication), „оторизация или възлагане на привилегии“ (Authorization) и „управление на потребителските сесии“ (Accounting) – “AAA”, използвайки Radius сървъра. AES е симетричен блоков криптографски подход, чиято дължина на ключа може да бъде 128, 192 или 256 bits. Изграждането на връзка между комутатора и безжичния рутер се базира на *Fast Ethernet* и *Ethernet* типове интерфейси.

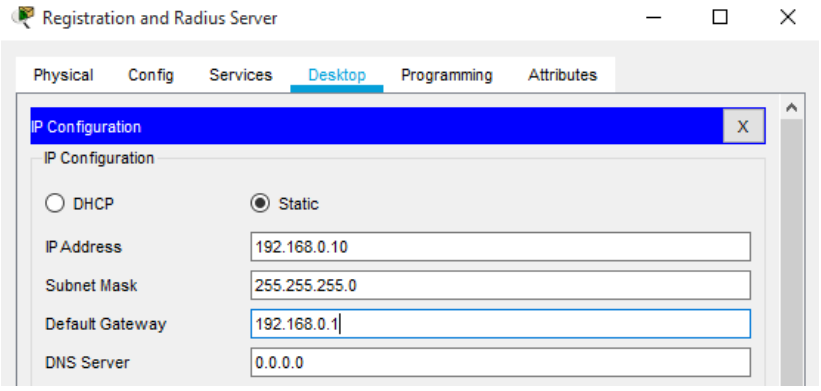
Дефинирана е следната последователност от действия за конфигуриране на потребителския достъп и мониторинг на IoT устройствата:

- ✓ Проверява се IP адрес на безжичния рутер чрез отваряне на конфигурационния GPU инструмент, указано на фиг. 14.28;



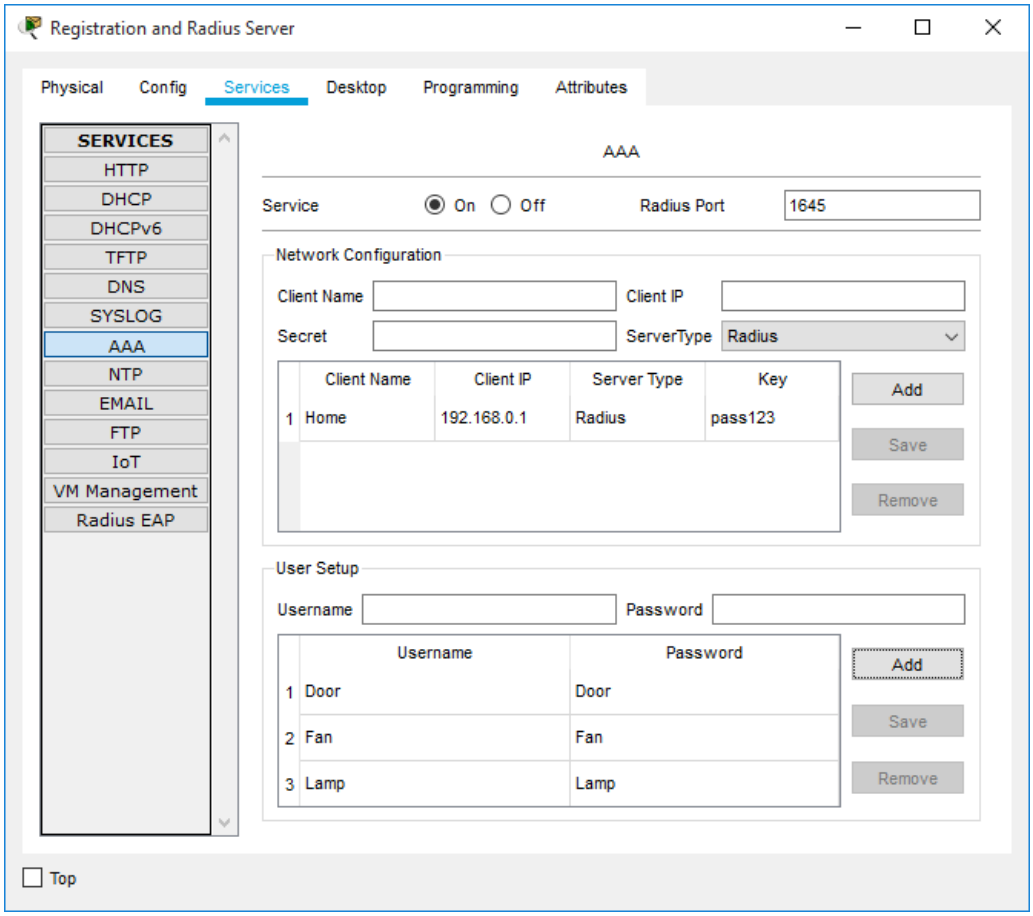
Фиг. 14.28. *GUI инструмент за назначаване на IP адрес и подмрежова маска на безжичен рутер WRT300N*

- ✓ Назначават се IP адрес, подмрежова маска и Gateway на сървърно звено Server-PT, както е дадено на фиг. 14.29;

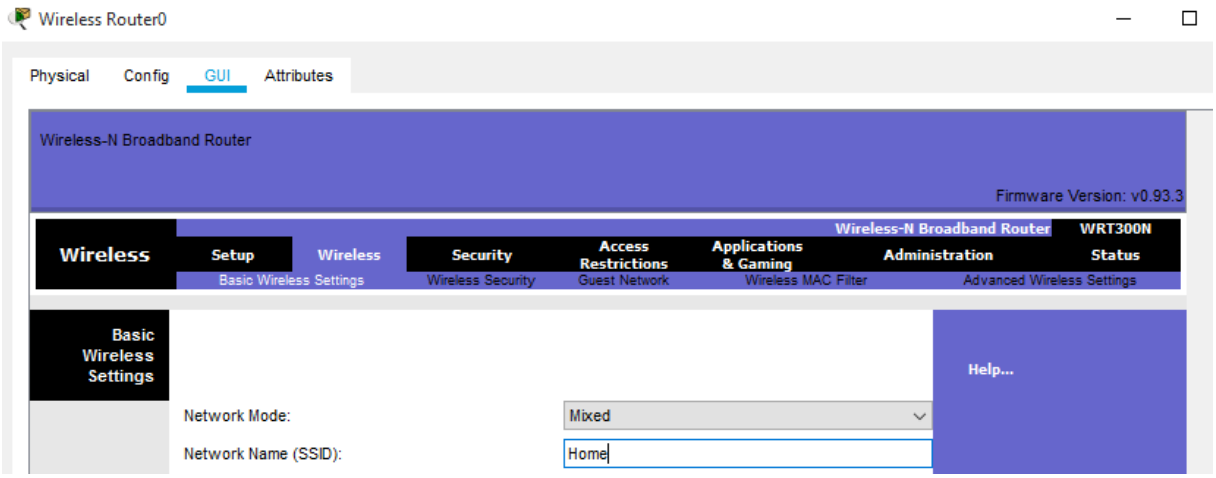


Фиг. 3.23. *Назначаване на IP address, Subnet mask и Gateway относно Server-PT*

✓ Дефинира се Radius Server от лентата с инструменти като се избират Services и услугата AAA. Активира се услугата и се извършва мрежова конфигурация на атрибутите “Client name”, “Client IP”, “Secret” и “Server type”, т.е. задава се име и IP адрес на клиента в случая мрежово устройство WRT300N, парола за сигурност и се избира тип на сървъра, съответно Radius. Дефинират се наличните IoT устройства, обект на отдалечен контрол. Стъпката е показана фиг. 14.30;

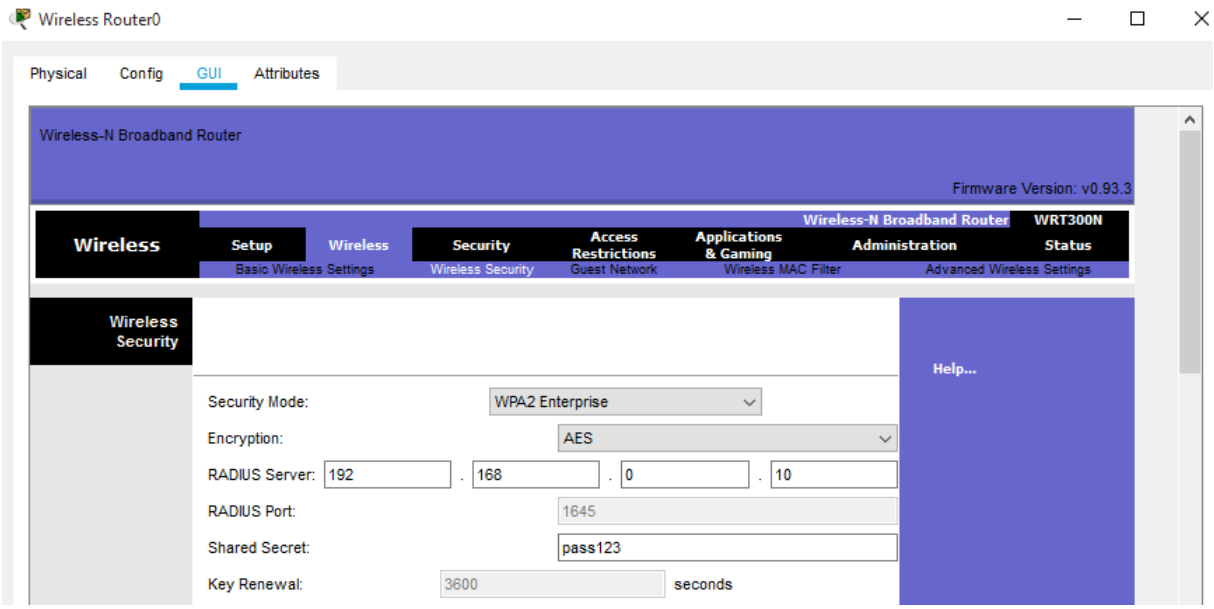


Фиг. 14.30. *Дефиниране на Radius Server и IoT устройства за мониторинг и контрол*



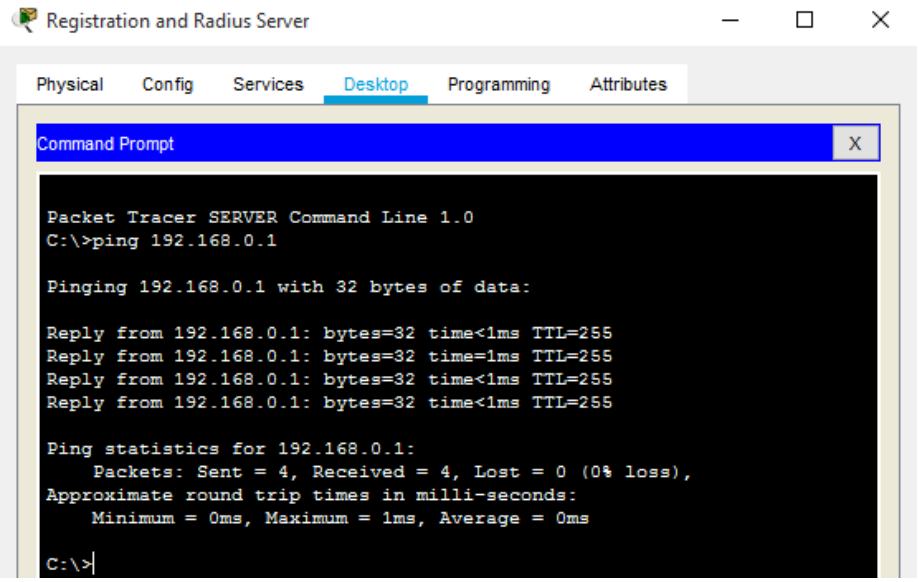
Фиг. 14.31. *Задаване на SSID на мрежата относно настройки на WRT300N*

✓ Конфигурира се безжичен рутер WRT300N като с идентификатор на безжичната комуникационна мрежа SSID “Home”, както е показано на фиг. 14.31. Избира се тип режим на сигурността на безжичната връзка “WPA2 Enterprise”, вид на криптиране “AES”. Въвеждат се настройки за достъп и връзка с Radius сървъра, като следва IP адрес и парола за сигурност. Стъпката е представена на фиг. 14.32;



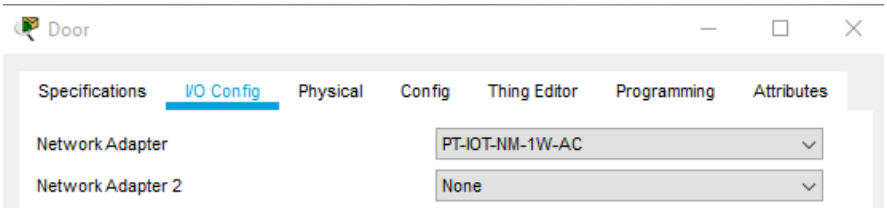
Фиг. 14.32. *Настройки на режим сигурност, метод на криптиране на връзката и достъпа до Radius сървъра относно конфигурация на WRT300N*

✓ Удостоверява се коректността на конфигурираната комуникационна връзка между Radius Server и безжичния рутер, онагледено на фиг. 14.33;

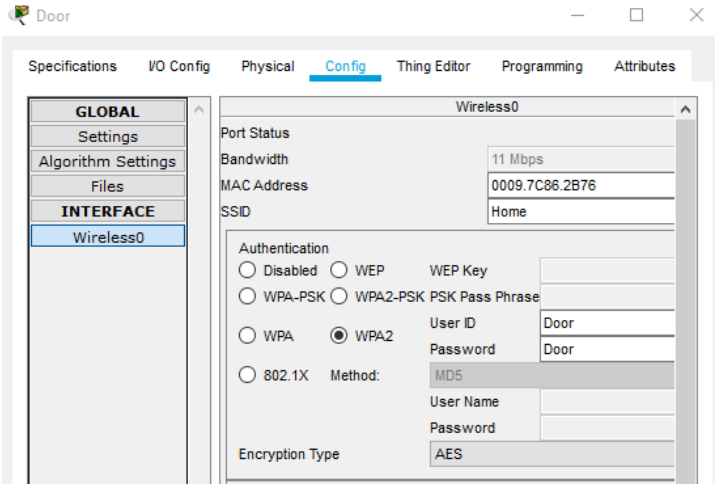


Фиг. 14.33. *Удостоверяване на връзката между Radius сървъра и WRT300N*

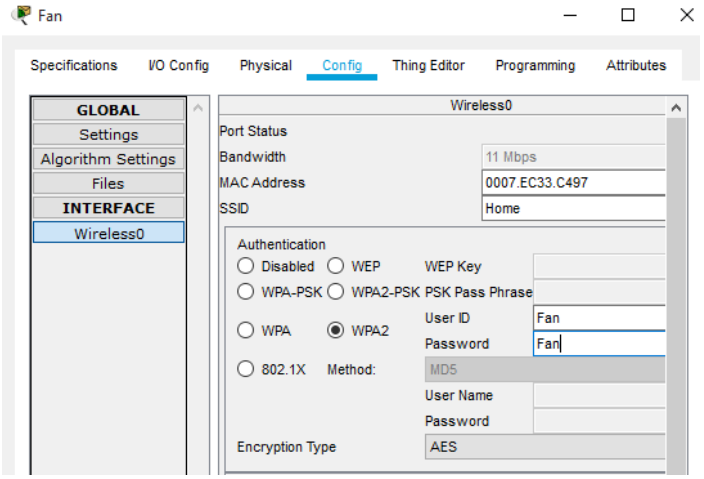
✓ Конфигурира се връзка между IoT устройствата и безжичния рутер като за целта за всяко устройство се избира тип на мрежовия адаптер PT-IOT-NM-1W-AC (фиг. 14.34). Задават се настройки на безжичния интерфейс SSID “Home” и се извършва автентикация – избор на режим на сигурност WAP2 и въвеждане на специфираните за всяко конкретно устройство “User ID” и “Password” (фиг. 14.35). След изпълнение на действието ще е налице свързаност между всички мрежови устройства, както се вижда на фиг. 14.36;



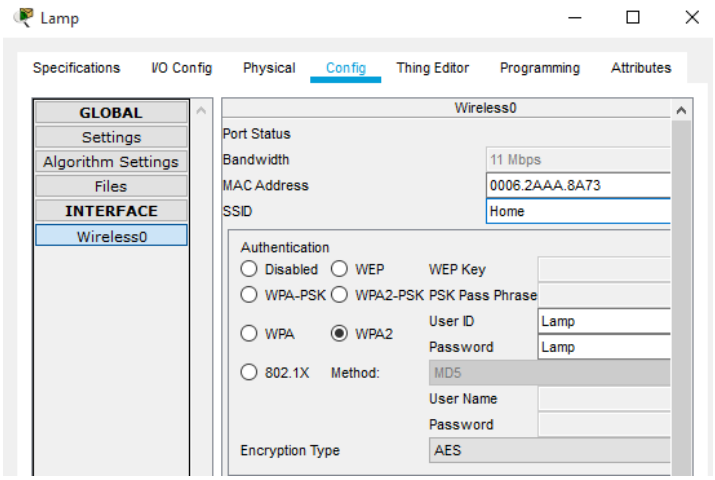
Фиг. 14.34. *Избор на мрежови адаптер PT-IOT-NM-1W-AC*



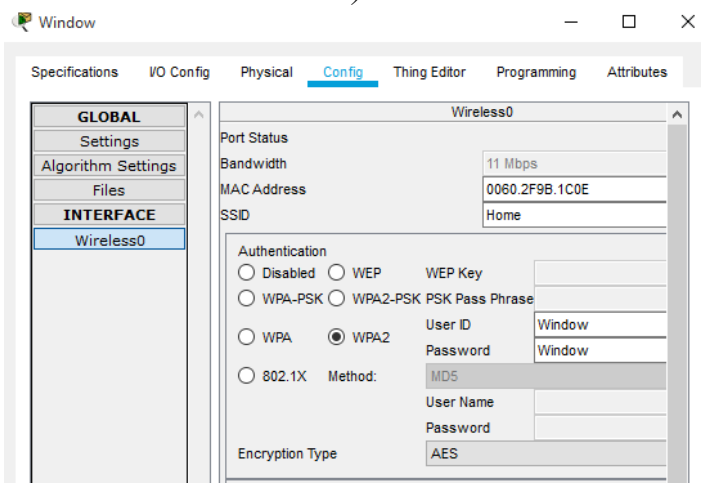
а)



б)

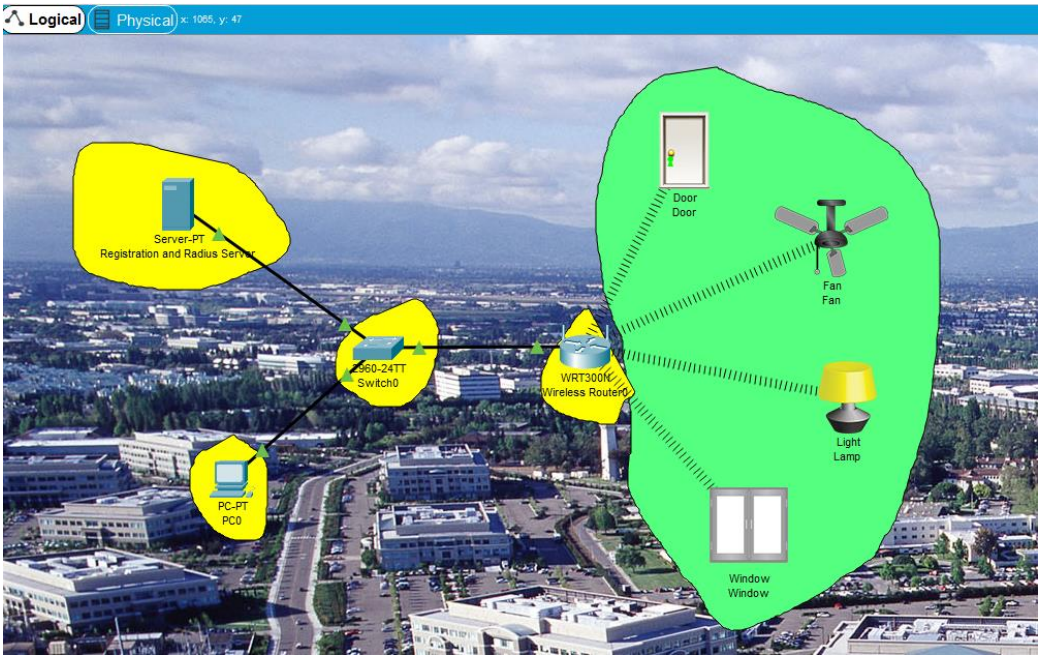


в)

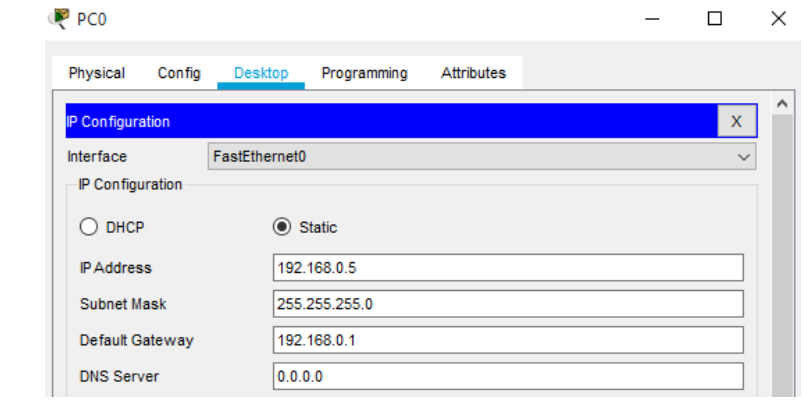


г)

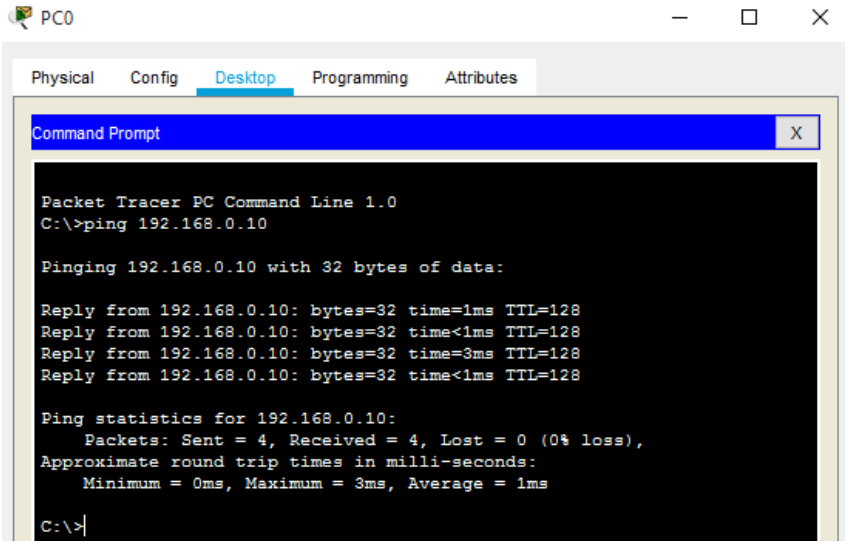
Фиг. 14.35. *Настройки на автентикация на IoT устройства а) Door, б) Fan, в) Lamp и г) Window*



Фиг. 14.36. *Завършен вид на система „Интелигенте дом“, базирана на Registration and Radius Server*



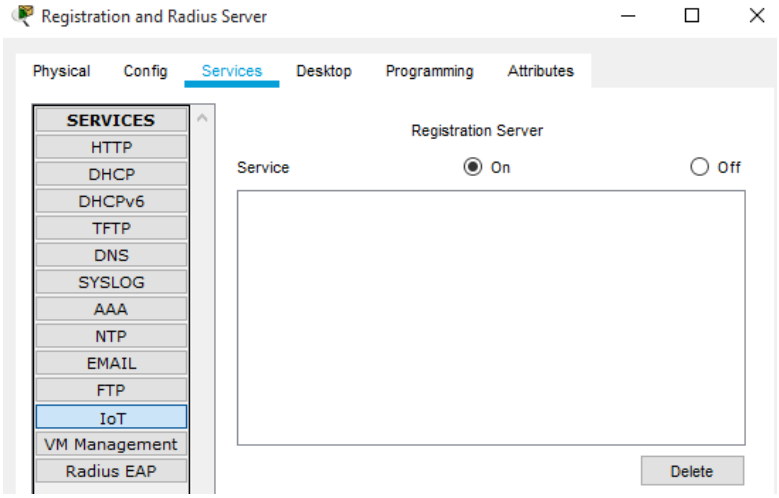
а)



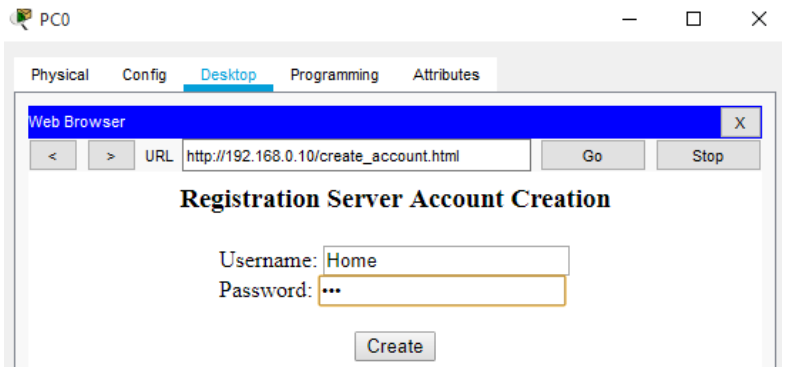
б)

Фиг. 14.37. *Настройки на PCO а) и проверка на комуникацията с Radius сървъра*

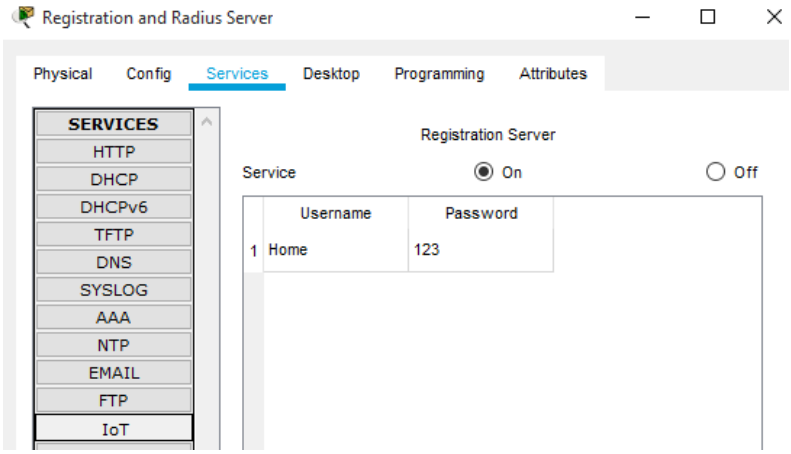
- ✓ Настройка на IP address, Subnet mask и Gateway относно потребителската компютърна станция проверка на комуникация с конфигурирания Radius сървър, както е показано на фиг. 14.37;
- ✓ Активиране на процедурата за назначаване на Registration Server от услуга “IoT” на сървърното звено, дадено на фиг. 14.38;



Фиг. 14.38. *Активиране на Registration Server*

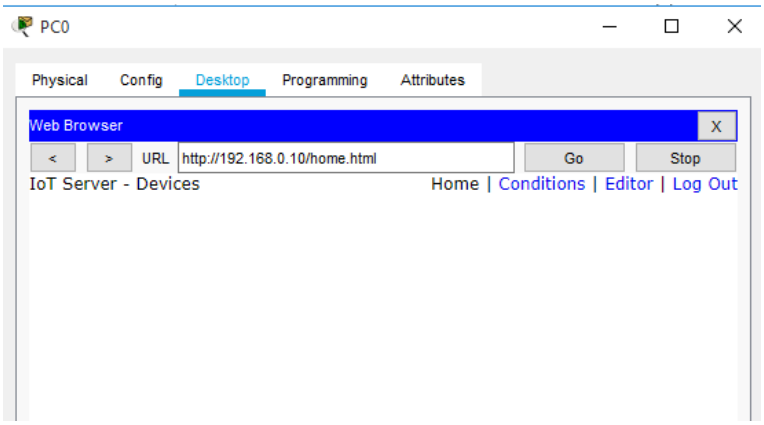


а)



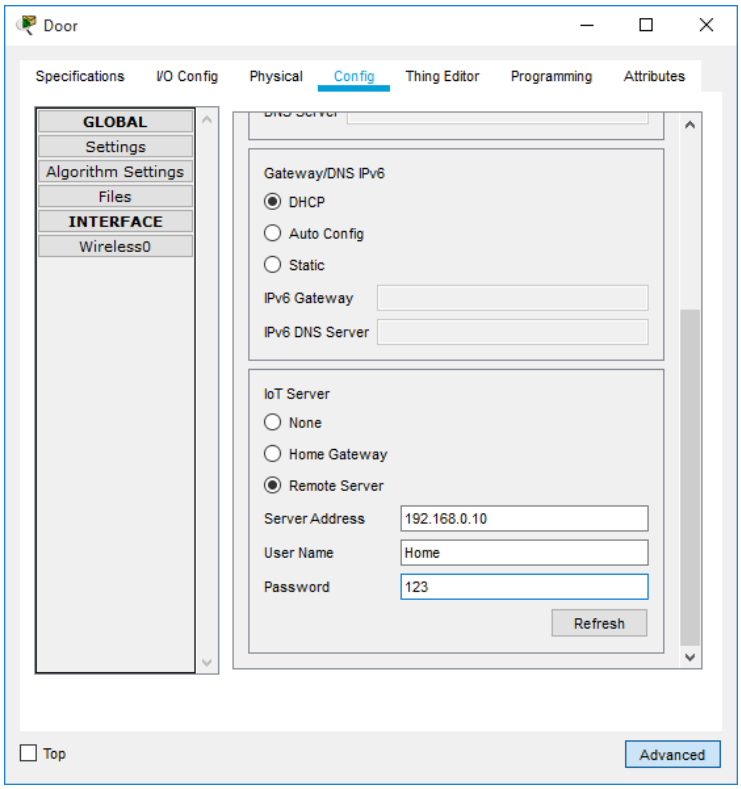
б)

Фиг. 14.39. *Създаване на потребителски профил а) и наличен създаден профил б) относно Registration Server*

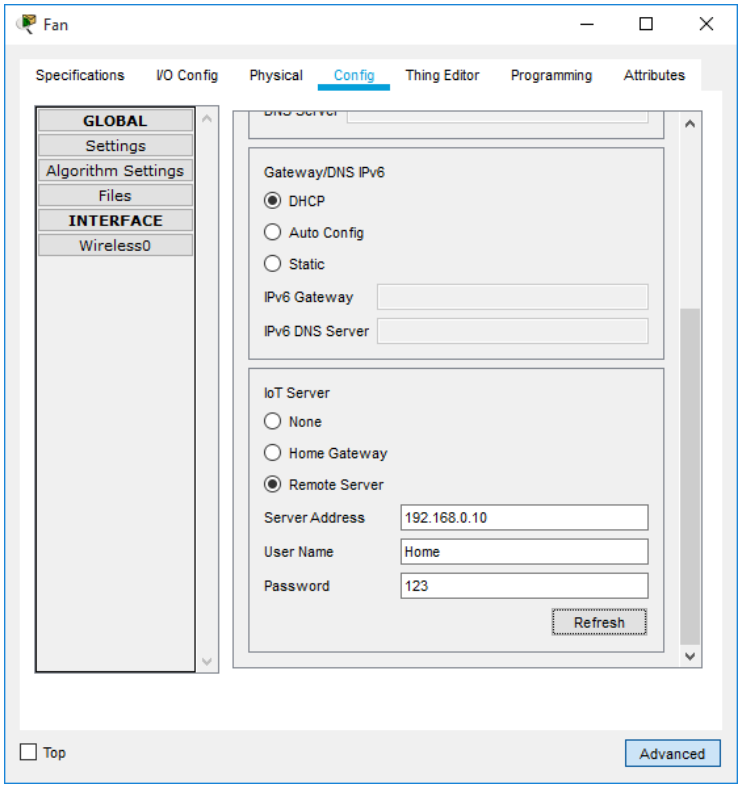


Фиг. 14.40. *Среда на Web приложение за мониторинг на IoT устройствата*

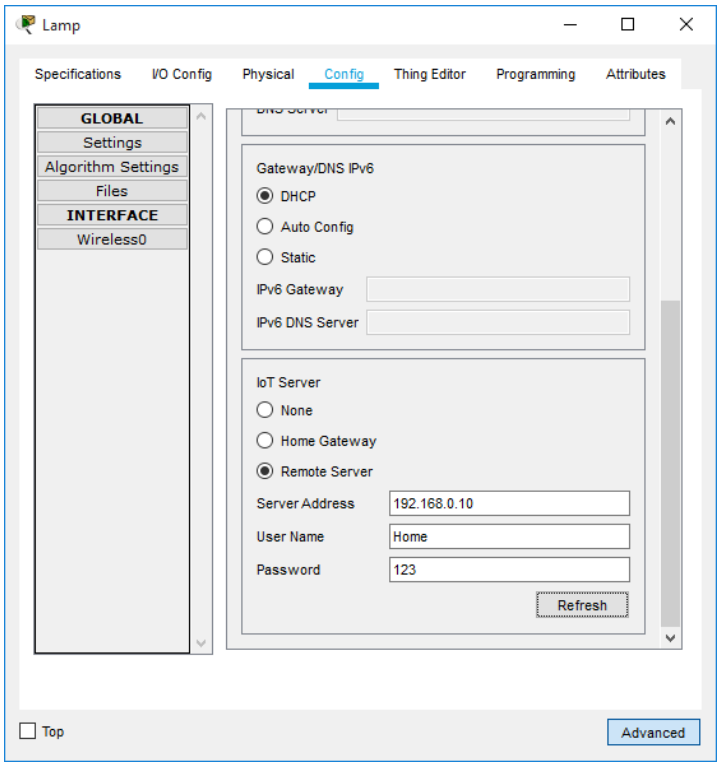
✓ Стартиране на Web Browser от потребителска станция PC0 и създаване на потребителски профил – избор на “Sing up now”, на Registration Server за видимост, достъп и управление на IoT устройствата, както е представено на фиг. 14.39. След създаване на профила в WEB приложението, достъпно от PC0 за мониторинг на устройствата, ще бъде налично (фиг. 14.40);


Door
Specifications
I/O Config
Physical
Config
Thing Editor
Programming
Attributes
GLOBAL
Settings
Algorithm Settings
Files
INTERFACE
Wireless0
Gateway/DNS IPv6
DHCP
Auto Config
Static
IPv6 Gateway
IPv6 DNS Server
IoT Server
None
Home Gateway
Remote Server
Server Address
192.168.0.10
User Name
Home
Password
123
Refresh
Top
Advanced

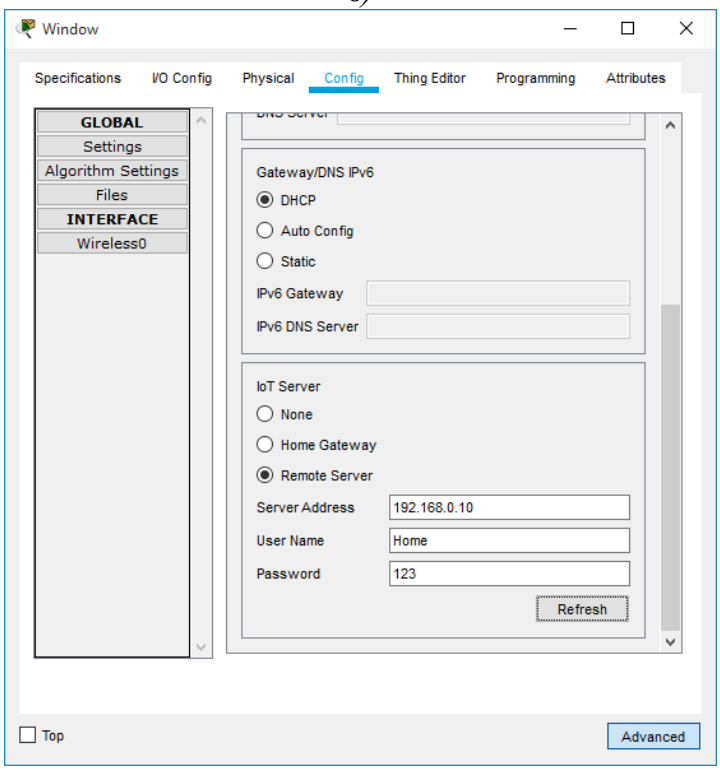
а)


Fan
Specifications
I/O Config
Physical
Config
Thing Editor
Programming
Attributes
GLOBAL
Settings
Algorithm Settings
Files
INTERFACE
Wireless0
Gateway/DNS IPv6
DHCP
Auto Config
Static
IPv6 Gateway
IPv6 DNS Server
IoT Server
None
Home Gateway
Remote Server
Server Address
192.168.0.10
User Name
Home
Password
123
Refresh
Top
Advanced

б)



в)



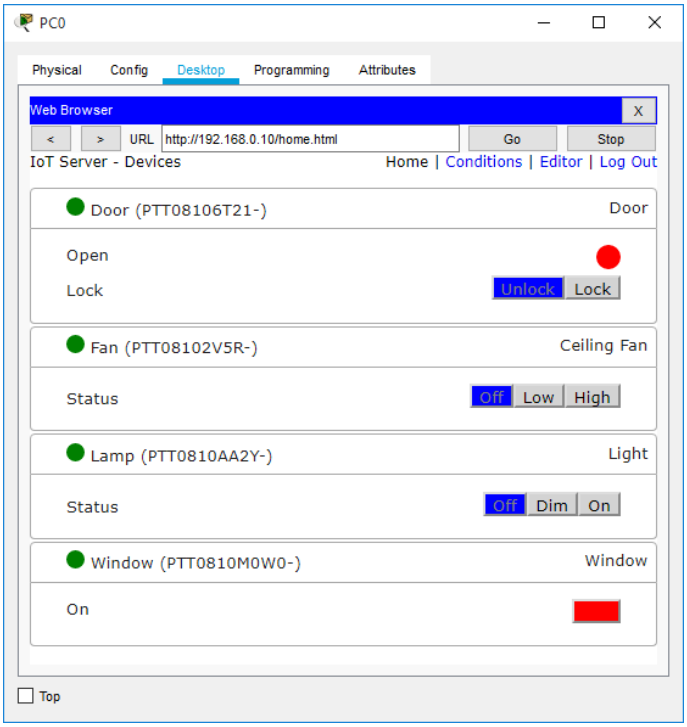
г)

Фиг. 14.41. *Задаване на адрес, потребителско име и парола на профила относно достъп до IoT устройства а) Door, б) Fan, в) Lamp и г) Window посредством Registration server*

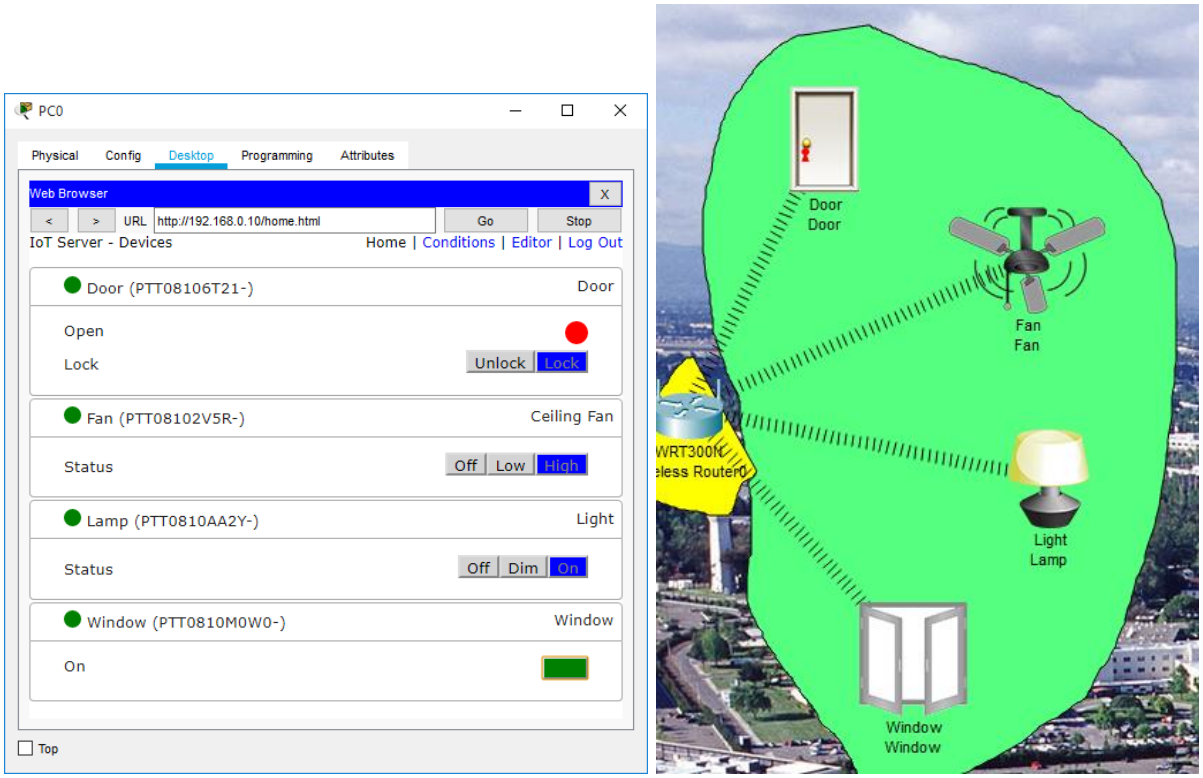
- ✓ За да бъдат видими IoT устройствата, следва задаване на IoT сървър тип “Remote Server” – указване на конфигурационни настройки по отношение на Registration Server като достъпен адрес, потребителско име и парола на

профила. Също така е необходимо активиране на връзката с Registration Server с избор на бутона Connect (фиг. 14.41);

✓ Удостоверяване на видимостта до устройствата през потребителската Web страница и упражняване на манипулации за отдалечено управление на IoT устройствата - фиг. 14.42 и фиг. 14.43.

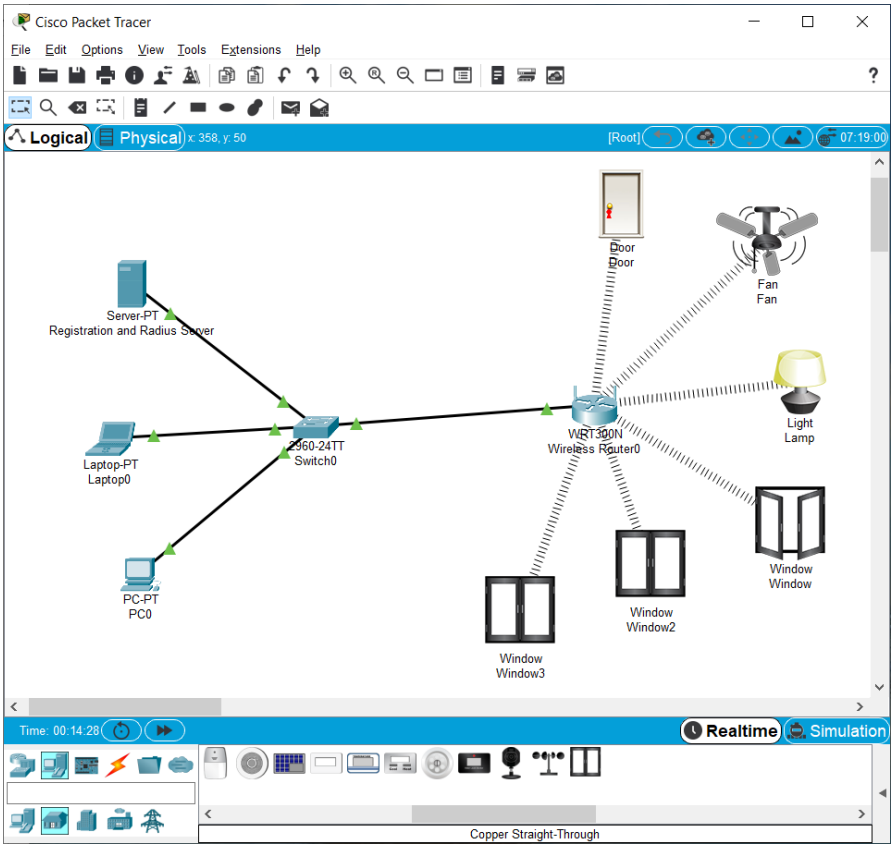


Фиг. 14.42. *Достъп и мониторинг до потребителските IoT устройства*

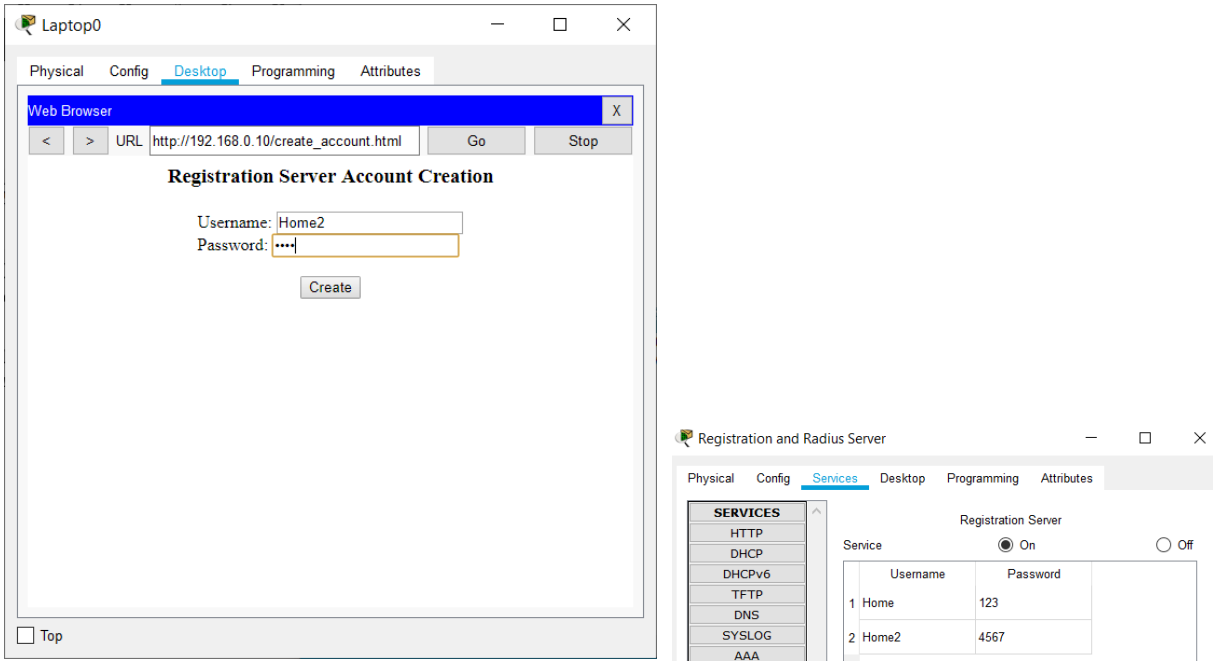


Фиг. 14.43. *Мониторинг и управление на IoT устройства*

✓ Разширяване на мрежовата топология с включване на втора потребителска станция Laptop0 и нови IoT устройства Window2 и Window3. Извършване на аналогични настройки спрямо указаните по-горе стъпки за конфигурация на функционирането на новите мрежови устройства. Модифицираният вид на IoT топологията е представен на фиг. 14.44.

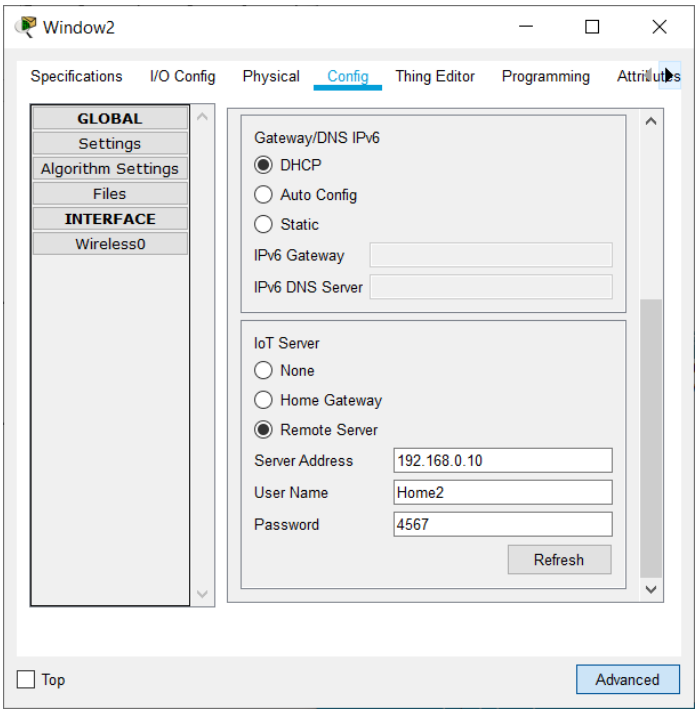


Фиг. 14.44. *Модифициран вид на системата „Интелигентен дом“*

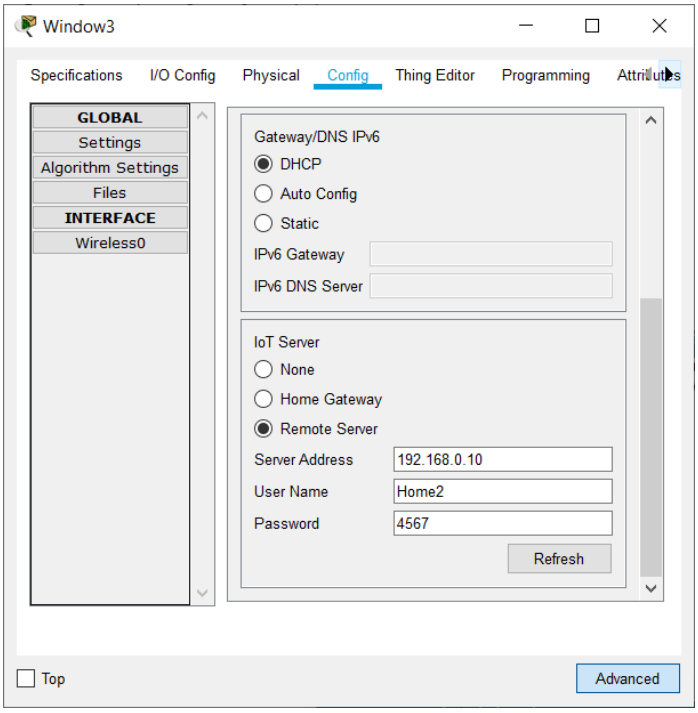


Фиг. 14.45. *Създаване на втори потребителски профил от станция Laptop0*

- ✓ създаване на втори потребителски профил от станция Laptop0 във връзка с достъп до и мониторинг на новите IoT устройства посредством Registration Server, както е дадено на фиг. 14.45;
- ✓ Провеждане на IoT Remote Server конфигурация на допълните IoT устройства във връзка достъпен адрес, потребителско име и парола на втория профил Home2 и активация на връзката с Registration Server чрез Connect, онагледено на фиг. 14.46;

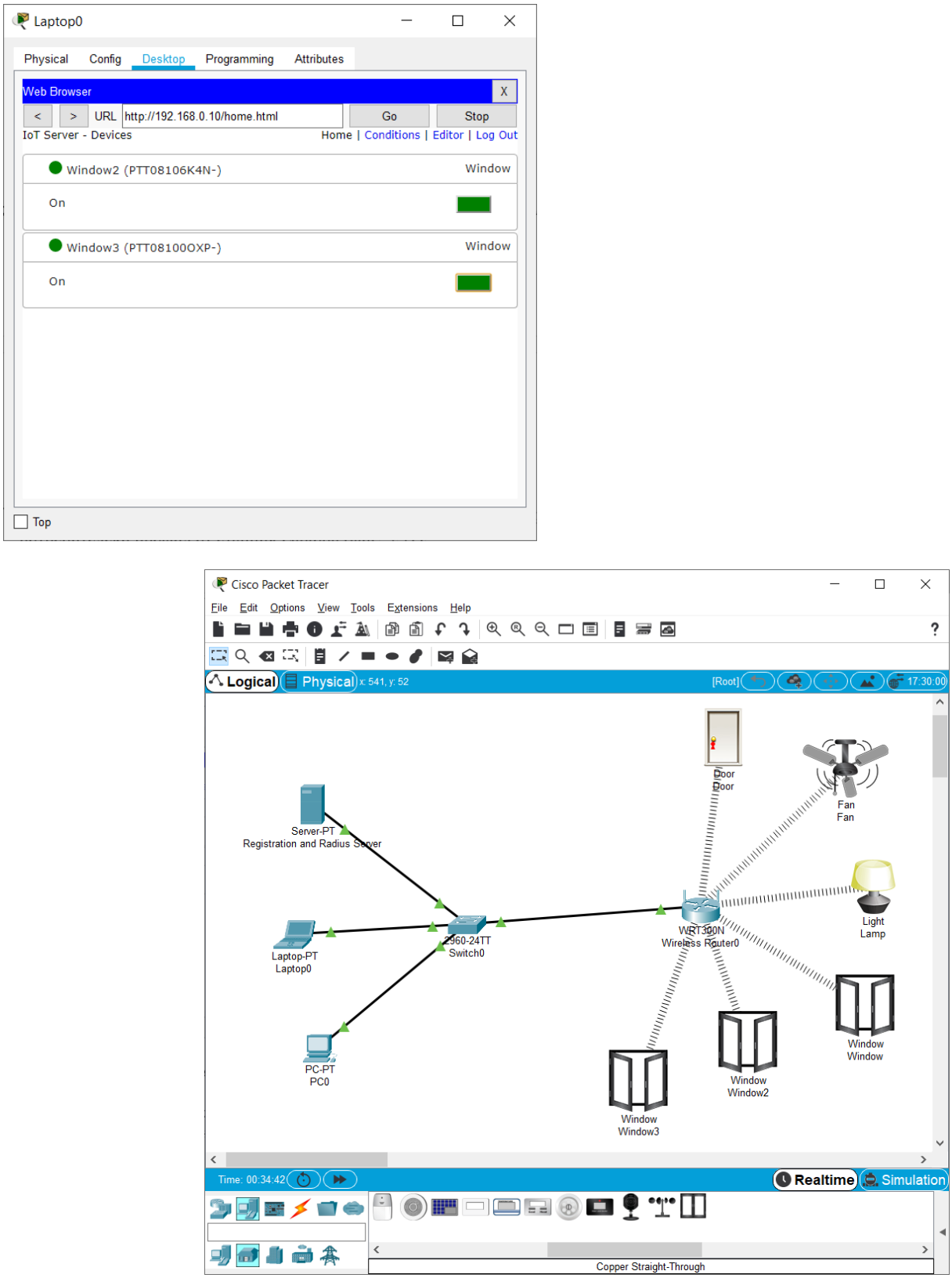


а)



б)

Фиг. 14.46. *Конфигурация на IoT Remote Server за IoT Window2 и Window3*



Фиг. 14.47. *Промяна на активността на IoT Window2 и Window3*

✓ Мониторинг и управление на състоянието на новите IoT устройства чрез втори потребителски профил от станция Laptop0 (фиг. 14.47).

14.4. Контролни въпроси

1. Кои са съвременните и бъдещи технилогии, свързани с внедряване и концептуално развитие на IoT технологията?
2. Какви операции по отношение на потребителския достъп се асоциират с услугата AAA?
3. Кои са възможните дължини на ключовете при AES криптиращия алгоритъм?
4. В какво се изразява предназначението на Radius сървъра във връзка със сигурността при изграждане на безжична мрежова свързаност?
5. Какви компоненти се оказват относно достъпа до IoT устройствата, които са обект на мониторинг в системите „Интелигентен дом“, при съответен режим за сигурност?

Лабораторно упражнение № 15

КОНТРОЛ НА ПОТРЕБИТЕЛСКИЯ ДОСТЪП ДО WEB И EMAIL СЪДЪРЖАНИЕ ПРЕЗ ОБЛАЧНИ СТРУКТУРИ В CISCO PACKET TRACER

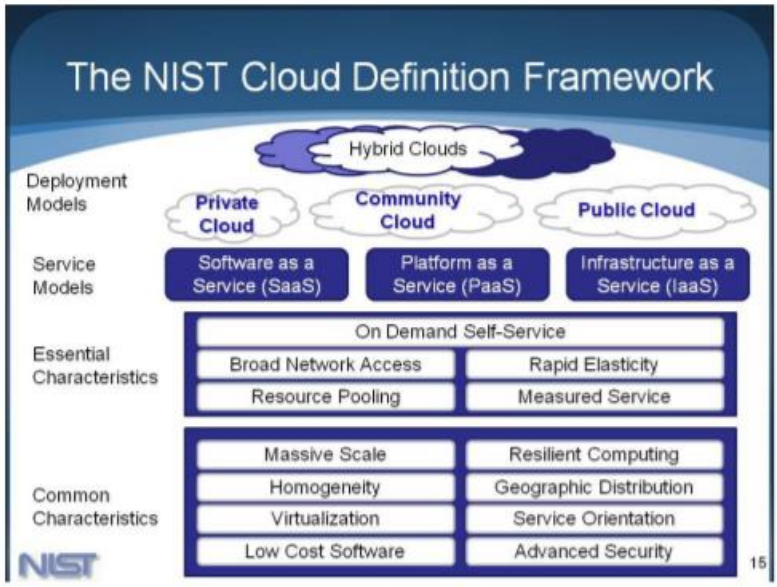
15.1. Цел на упражнението

Да се запознаят студентите с подходите за управление на сигурността при достъп до WEB съдържание и E-mail потребителски съобщения чрез оторизация на създадени домейни и потребителски профили през облачни обекти в симулационна среда Cisco Packet Tracer.

15.2. Теоретична постановка

Концептуален модел на облачните технологии

Облачните технологии се състоят от разнородни типове услуги, които са на разположение на потребителите при поискване (on-demand). Според концептуалният модел на облачните технологии, даден на фиг. 15.1, три са основните модели на облачни услуги, респективно:



Фиг. 15.1. *Модел на облачните технологии*

- *Софтуер като услуга - Software as a Service (**SaaS**)* – При този модел доставчикът предоставя на клиентите достъп до лицензирани софтуерни приложения, които са инсталирани на облака;

- *Платформа като услуга – Platform as a Service (**PaaS**)* – Доставчиците предоставят на клиентите програмен език като платформа или софтуер като например Java, Python или .Net (не само тези, разбира се), за да внедрят създадени собствени или приложения на трета страна в облачната инфраструктура и да ги направят достъпни през интернет с API (Application Program Interfaces) или уебсайт портали за своите клиенти;

- *Инфраструктура като услуга – Infrastructure as a Service (**IaaS**)* – Доставчиците на облачна инфраструктура като услуга предоставят на клиентите възможност да ползват изчислителни възможности, дисково

пространство, интернет мрежа, оперативна памет и други основни технологични ресурси, които правят възможно внедряването и работата на различни софтуерни програми като операционни системи и приложения.

Мерки за защита на личните данни в „облака“

Съществуват и редица международни документи, приложими в сферата като напр. Насоки за стандартни европейски SLA (т.е. споразумения за ниво на качество на услугата при предоставяне на „облачни“ услуги (Cloud Service Level Agreement Standardization Guidelines), както и различни стандарти и ръководства на организации като ENISA, NIST, ISO/IEC7. Тези документи могат да са в помощ както на доставчиците, така и на ползвателите на „облачни“ услуги при уреждане на отношенията им. В допълнение, през ноември 2012 г. Работната група по чл. 29 от Директивата за защита на личните данни прие становище (Становище №05/2012) относно „cloud computing”. Работната група по чл. 29 е консултативен орган и няма законодателни правомощия. Издаваните от нея становища имат тълкувателен характер и отразяват възгледите на европейските органи по защитата на личните данни относно прилагането на нормативната уредба. В тази връзка, набелязаните в Становище №05/2012 изисквания следва да бъдат съобразени, доколкото не противоречат на местното законодателство на съответните администратори [29].

а)

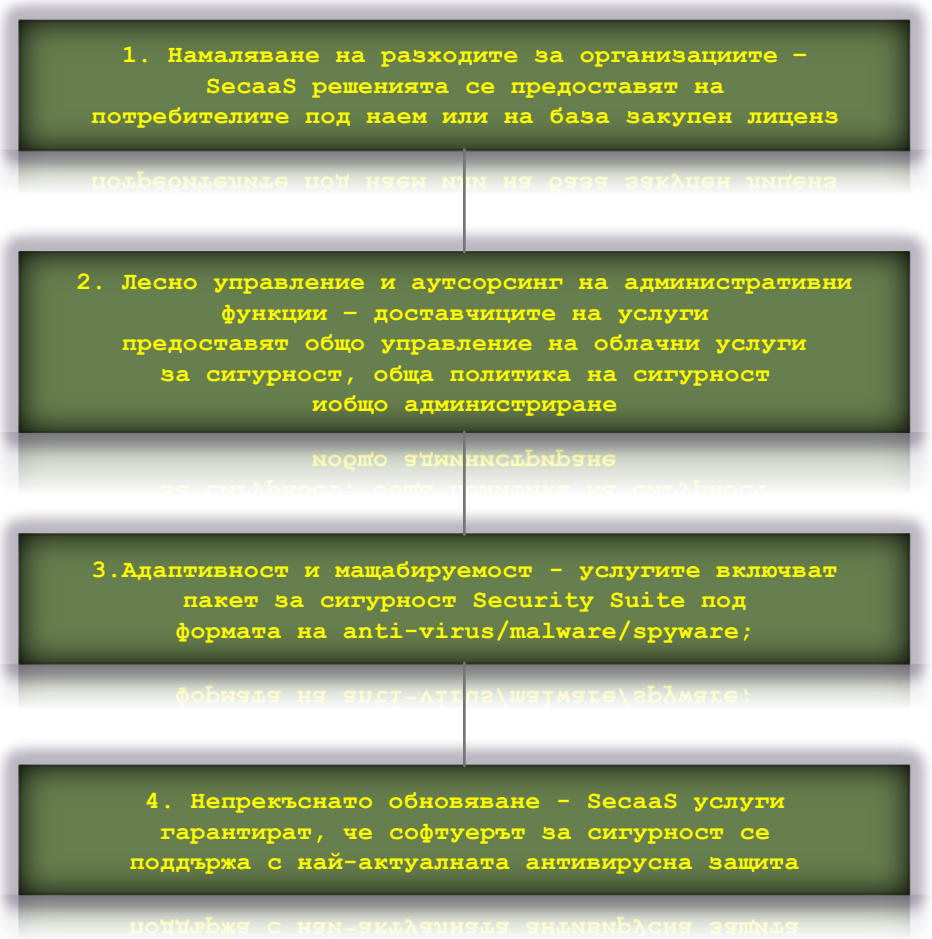
б)

Фиг. 15.2. *По-важни изисквания, набелязани от Работната група по чл. 29 – а) част №1 и б) част №2*

В Становище №05/2012 подробно се разяснява какви конкретни действия и мерки е необходимо да предприеме един администратор, използващ „облачни“ услуги, за да се счете, че съблюдава изискванията на уредбата относно достъпа и обработката на личните данни. Същите са базирани основно на конструкция, при която отношенията „клиент-доставчик“ са отношения между „администратор и обработващ“. Подчертава се, обаче, че е възможно в определени случаи доставчикът на „облачни“ услуги да действа и като администратор, ако обработва допълнително лични данни, предоставяни му от клиента за свои собствени цели. Най-съществените изисквания, набелязани от Работната група по чл. 29, са обобщени на фиг. 15.2 [30].

Сигурността като облачна услуга – Security as a service

Сигурността като услуга (security as a service), известна като съкращението ***SecaaS (или SaaS)*** е модел на облачна технология, при който доставчикът осигурява управлявани услуги за сигурност на потребителите в Интернет. ***SecaaS*** е базиран на облачната технология, софтуер като услуга SaaS (software as a service), но се ограничава до специализирани услуги за информационна сигурност на данните и приложенията на потребителите. Според Techopedia ***SecaaS*** „улеснява осигуряването на управлявани услуги за сигурност в облака, като носи редица ползи на организациите“, описани на фиг. 15.3.



Фиг. 15.3. *Ползи от* ***Secaas*** *(Сигурността като услуга)*

Важно значение при ползването на облачните услуги има осигуряването на защита на информацията, което е свързано с идентифицирането на заплахите и набелязване на възможните решения за осигуряване на информационната сигурност на данните и приложенията на потребителите в облачната среда и насочване на усилията за преодоляването им в конкретни направления. Според редица организации като:

- ***Cloud Security Alliance (CSA)***;
- ***Cloud Computing Information Assurance Framework (ENISA)***;
- ***Information Security Forum (ISF)*** и т.н.,

част от основните ***заплахи за информационната сигурност*** на потребителите в облачната среда са свързани с процесите, описани на фи. 15.4 [29, 30].

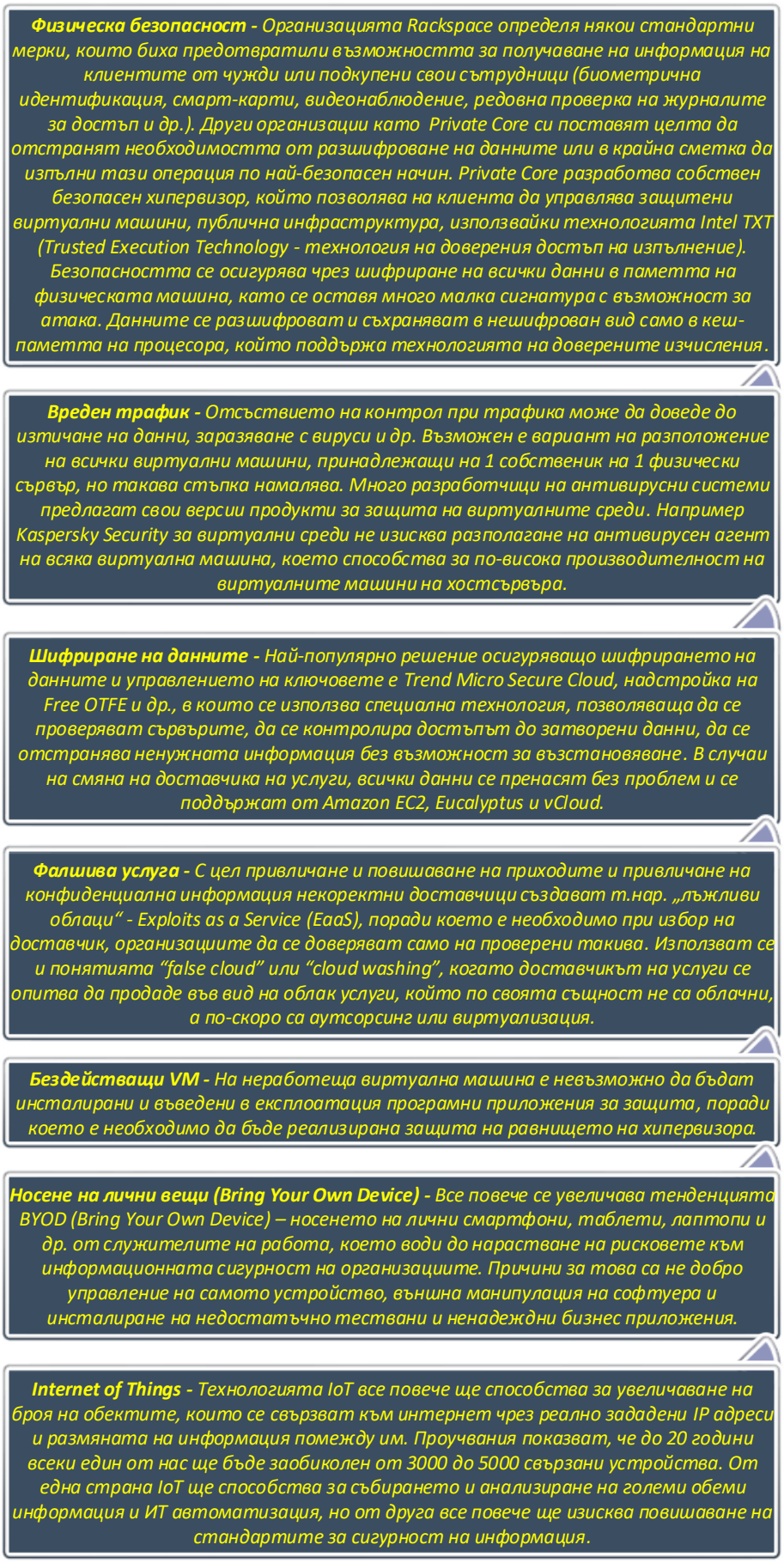
Оторизация на потребителя - При ползване на облачни услуги най-широко разпространен способ за оторизация на потребителите е защита с парола. Доставчиците осигуряват на потребителите по-висока надеждност, като прилагат и средства като сертификати и електронни подписи. Прилагат се стандартите като LDAP (Lightweight Directory Access Protocol), SAML (Security Assertion Markup Language) и др. Потребителят може да се включи към виртуално работно място посредством IaaS, чрез използване на устройства като смартфон и др. Разграничаването на пълномощията (separation of duties) и контрола на всички събития е много важен, тъй като освен пряко увреждане на данните, може да има и кражба на данни.

Отделяне на данните на потребителите - *Най-добър вариант е когато всеки потребител използва индивидуална виртуална машина (Virtual Machine – VM) и виртуална мрежа. Разделението на VM между потребителите се осигурява от хипервизори (хипервайзор или монитор на виртуални машини е компютърен софтуер, фърмуер или хардуер, който създава и управлява виртуални машини), а разделението между виртуалните мрежи се осигурява чрез прилагането на стандартни технологии като VLAN (Virtual Local Area Network), VPLS (Virtual Private LAN Service) и VPN (Virtual Private Network).*

Шифриране на данните - *Най-популярно решение, осигуряващо шифрирането на данните и управлението на ключовете е Trend Micro Secure Cloud, надстройка на Free OTFE и др., в които се използва специална технология, позволяваща да се проверяват сървърите, да се контролира достъпът до затворени данни, да се отстранява ненужната информация без възможност за възстановяване. В случаи на смяна на доставчика на услуги, всички данни се пренасят без проблем и се поддържат от Amazon EC2, Eucalyptus и vCloud.*

Hyper jacking - *Една от специфичните хакерски атаки е получила името "hyper jacking", което означава подмяна на хипервизорите с такива, които позволяват нерегламентиран достъп до предлаганите услуги. В същност заплахата за хипервизорите си остава хипотетична, тъй като за реализация на атака срещу тях е необходима висока квалификация, физически достъп до сървъра, а също и процесор, който да поддържа виртуализация на апаратно равнище. Microsoft работят съвместно с водещи производители на процесори като Intel и AMD с цел разработване на средства за апаратна защита (шифриране на процесорно равнище) и намаляване опасността от hyper jacking.*

а)

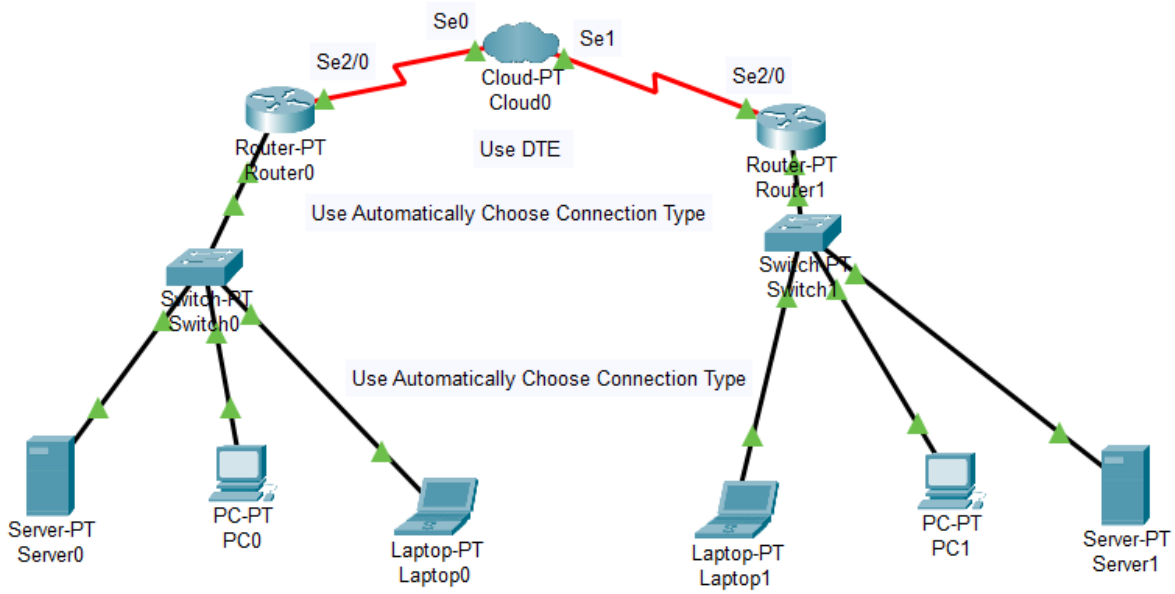


б)

Фиг. 15.4. *Заплахи за сигурността в облачна среда – а) част №1 и б) част №2*

15.3. Последователност на работа и задачи за изпълнение.

Осигуряването на информационна сигурност по отношение на достъп и преглеждане на WEB съдържание, изпращане и получаване на Email съобщения е важна задача в съвремнните комуникации. Обслужваният потребитекски WEB и Email трафик може да бъде направляван през различни типове преносни среди като една възможностите за това е използването на Cloud технологията. В тази секция се разглежда подобен тип задача, чиято мрежова реализация е представена на фиг. 15.5.

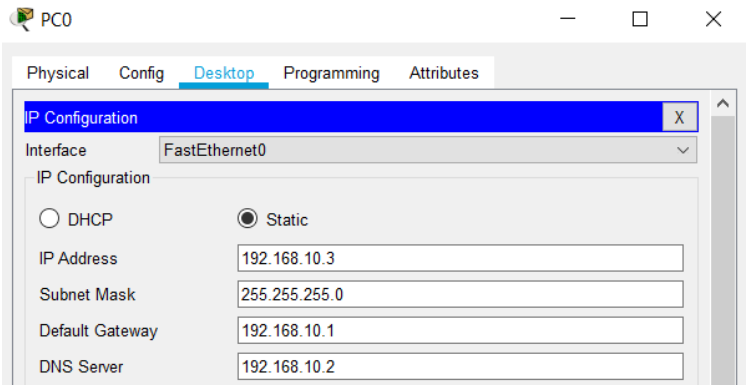


Фиг. 15.5. *Cloud инфраструктура с управление на WEB и Email достъпа по отношение на потребителските профили*

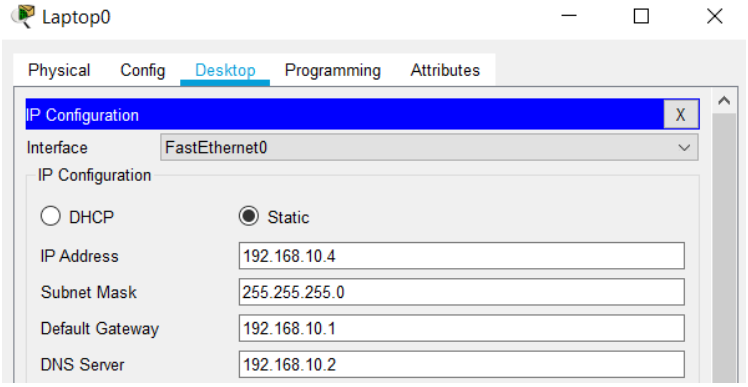
В връзка с обектната мрежова архитектура с управление на потребителския достъп е систематизирана следната стъпкова последователност от конфигурационни действия, респективно:

- **<u>Фаза №1: Конфигурация на мрежовите устройства</u>**

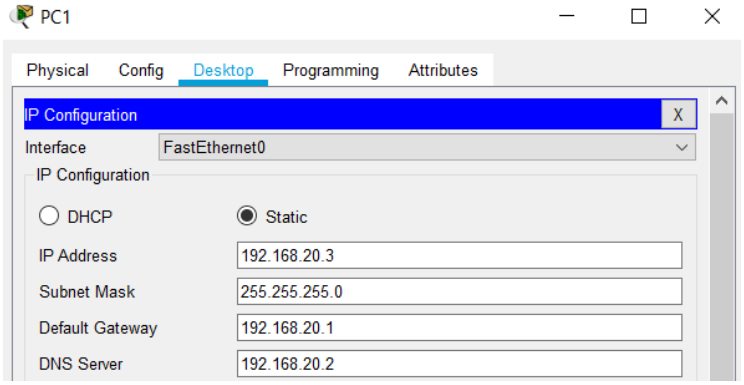
❖ Настройки на компютърните потребителски станции, сървърните станции и маршрутизаторите, състоящи се в назначаване на IP адрес, подмрежова маска, Gateway по подразбиране и DNS сървър. На фиг. 15.6 е дадена конфигурацията на мрежови устройства PC0, Laptop0, PC1 и Laptop1, докато на фиг. 15.7 на сървърните звена. За мрежовите маршрутизатори се въвеждат IP address Subnet Mask за комуникационния Fast Ethernet интерфейс, както е посочено на фиг. 15.8;



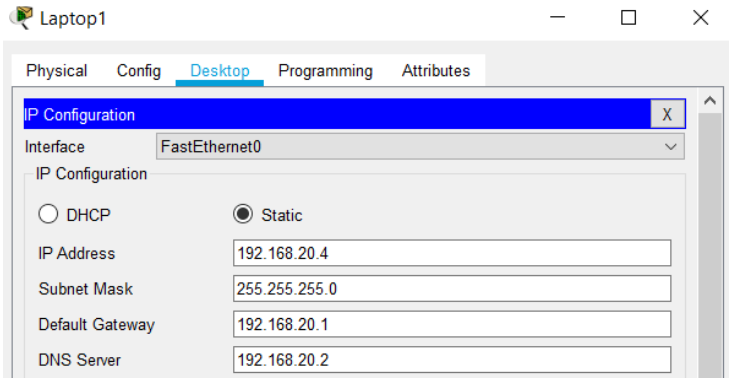
а)



б)

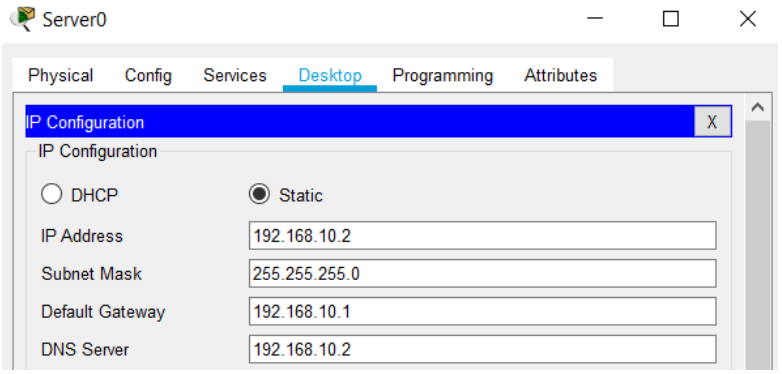


в)

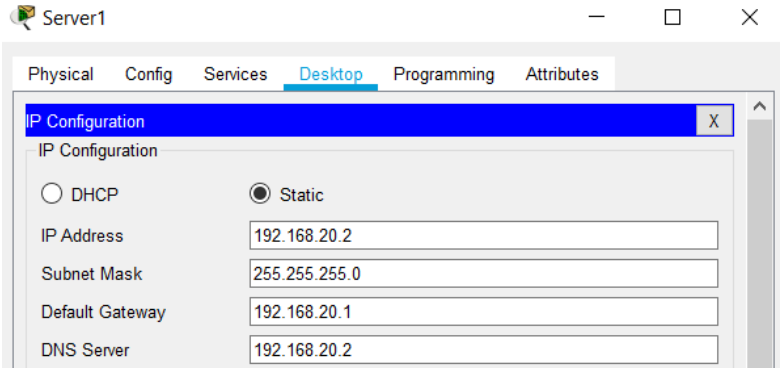


г)

Фиг. 15.6. *Конфигурационни параметри на мрежови устройства а) PC0, б) Laptop0, в) PC1 и г) Laptop1*

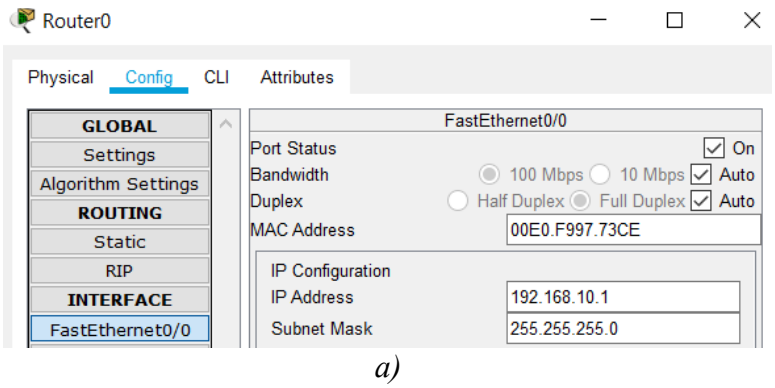


а)

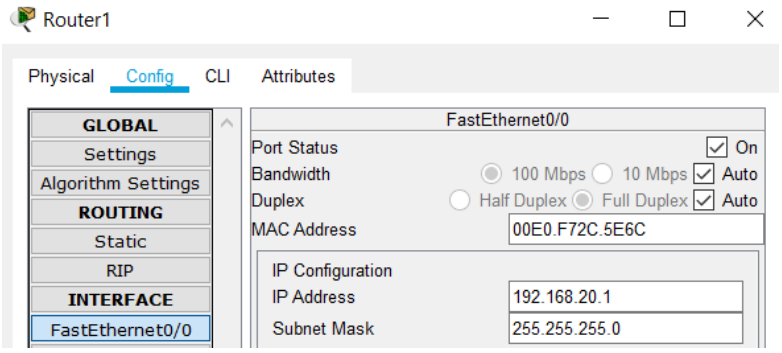


б)

Фиг. 15.7. *Конфигурационни параметри на мрежовите сървъри а) Server0 и б) Server1*

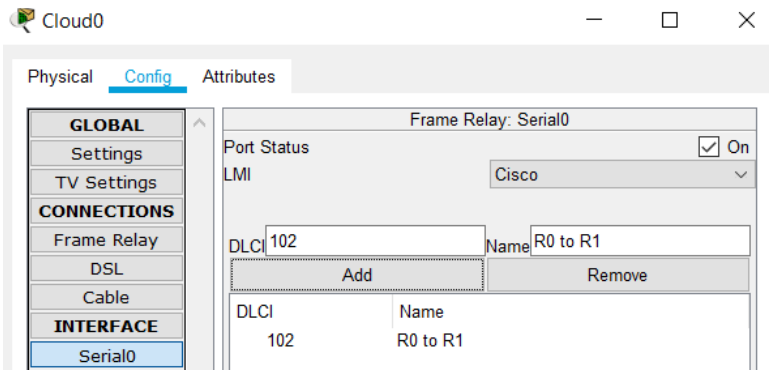


а)

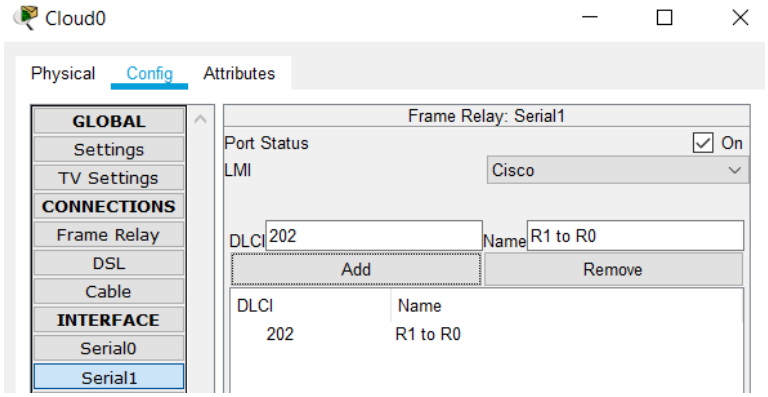


б)

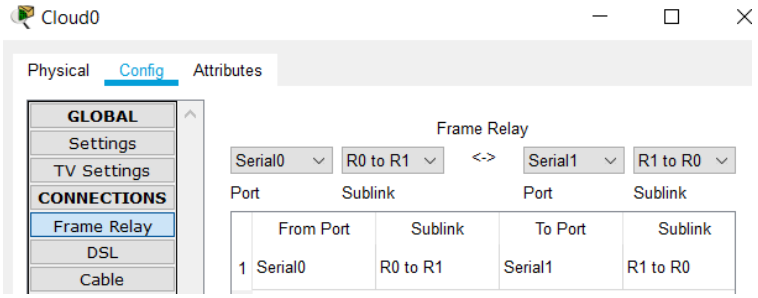
Фиг. 15.8. *IP конфигурация на маршрутизатори а) Router0 и б) Router1*



а)



б)

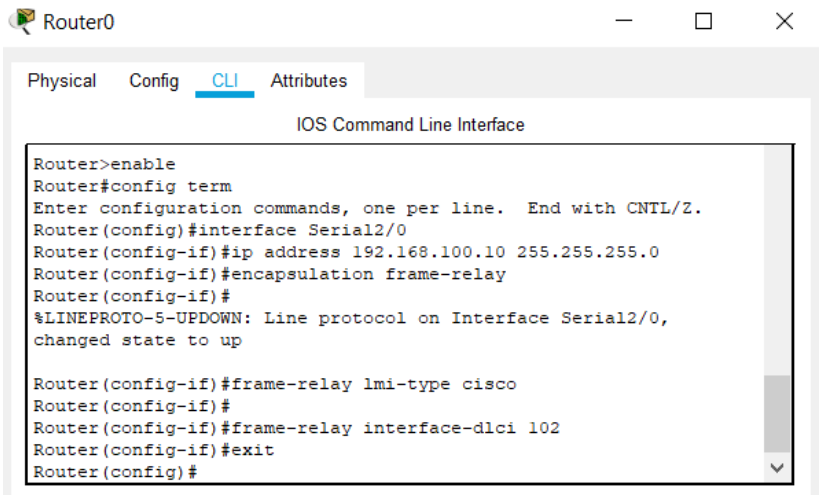


в)

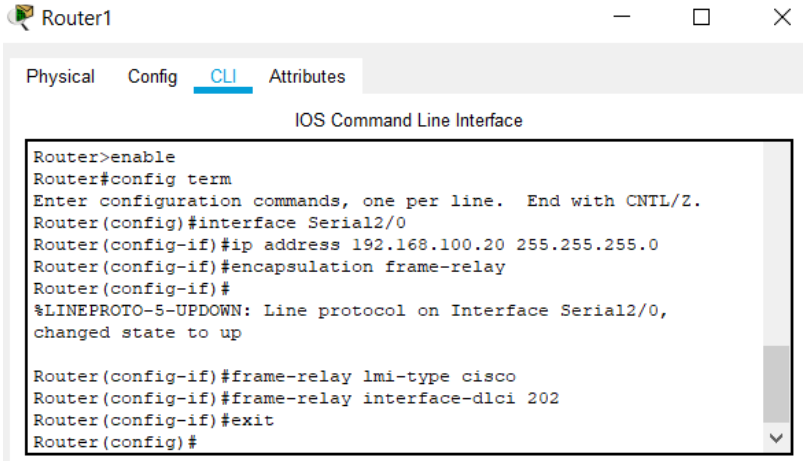
Фиг. 15.9. *Cloud конфигурация на а) Serial0, б) Serial1 и в) Frame Relay*

- Конфигурация на ***облачната структура*** по отношение на серийните интерфейси, както и типа на връзката Frame Relay, илюстрирано на фиг. 15.9. Относно използваните Serial0 и Serial1 комуникационни интерфейси се задава уникален DCLI (Data-Link Connection Identifier) идентификатор на връзката. Също така се извършва дифиниране на двупосочното напраление на пакетно предаване на данни от и към мрежовите маршрутизатори в обхвата на Frame Relay мрежовия сегмент;

- Настройки във връзка с Frame Relay стандарта за капсулиран пакетен пренос на данни между потребителски устройства, които са специфицирани в категорията терминално оборудване, при серийните интерфейси Serial2/0 на основните маршрутизатори - фиг. 15.10. За двата серийни интерфейса се назначават IP address и Subnet Mask. При Frame Relay се указват „интерфейс за управление на локалната мрежа“ LMI (Local Management Interface) и назначените DCLI индентификатори. Действията се изпълняват в Command Line Interface средата на Router0 и Router1;



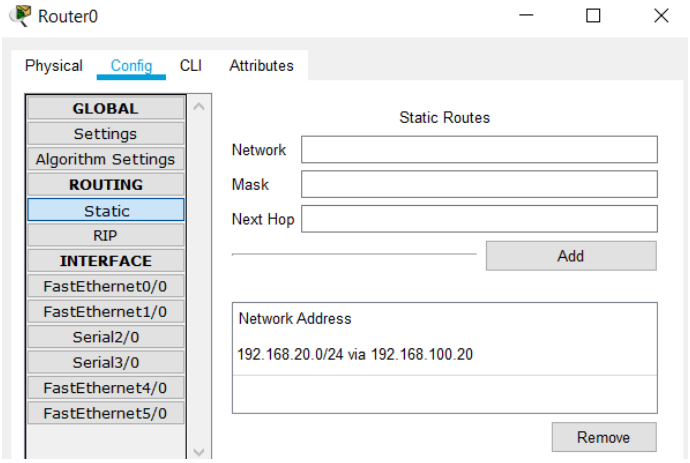
а)



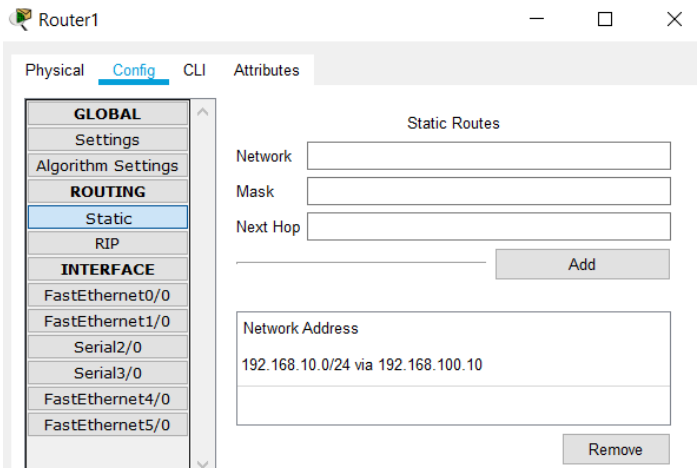
б)

Фиг. 15.10. *Frame Relay настройки относно Serial2/0 интерфейси при а) Router0 и б) Router1*

- Допълнителни настройки на мрежовите рутери на статичен тип маршрутизация Static – адрес на мрежата Network, мрежова маска Mask (255.255.255.0) и „следващ скок“ Next Hop (адресът на следващо устройство, към което ще бъдат изпратени пакетни данни), показано на фиг. 15.11.

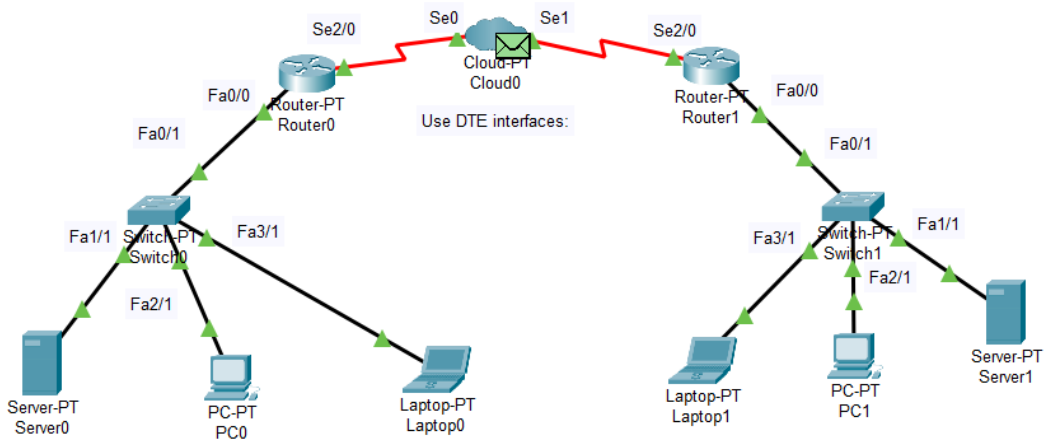


а)



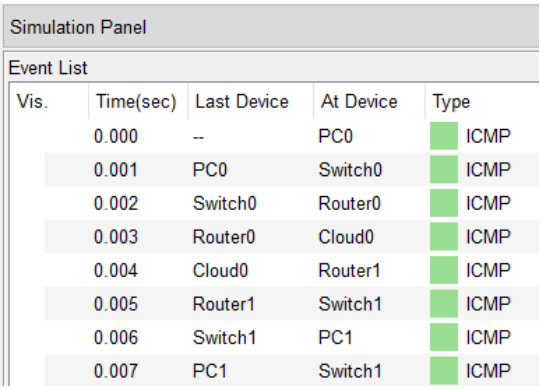
б)

Фиг. 15.11. *Настройки на статичен тип маршрузация на пакети при интерфейси при а) Router0 и б) Router1*



Фиг. 15.12. *Изпращане на PDU единици между PCO и PC1 в симулационен режим на работа*

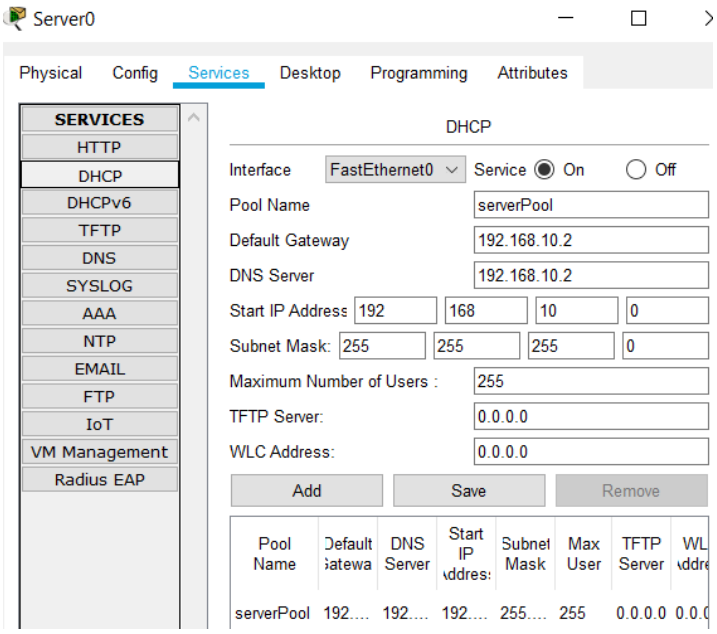
- Междинно тестване на пакетното предаване на данни чрез проверка с PDU информационни единици (указани на мрежовите устройства в Realtime Mode) между компютърни станции PCO и PC1 в Simulation Mode, показано на фиг. 15.12. Направлението на информационния поток може да бъде видяно на фиг. 15.13;



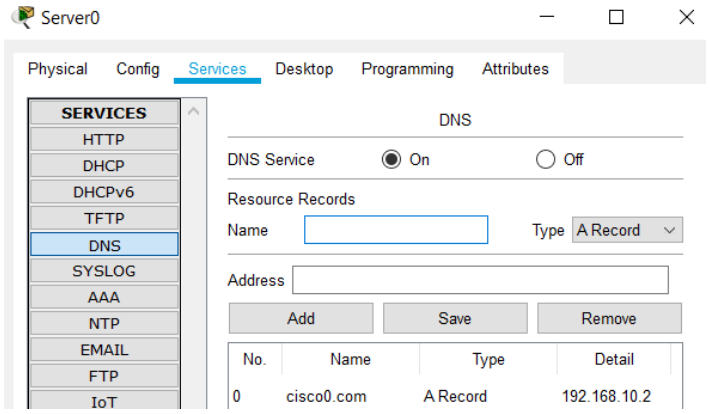
Фиг. 15.13. *Прослядаване на пакетите между PCO и PC1 в симулационен режим на работа*

- **Фаза №2: Управление на WEB достъпа**

❖ Конфигуриране на DHCP – активация на услугата, фиксиране на Default Gateway и DNS Server, и DNS – активация на услугата, въвеждане на домейн име и IP адрес, услуги при сървърните станции Server0 и Server1. Действията от процедурите са представени на фиг. 15.14 и фиг. 15.15;

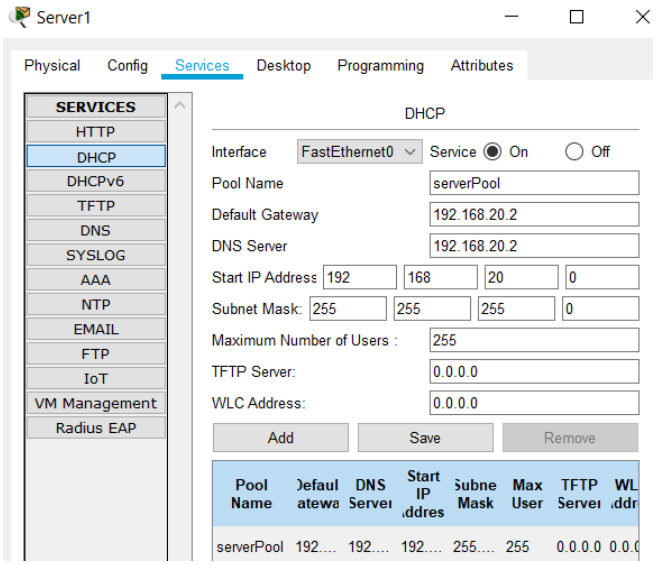


а)

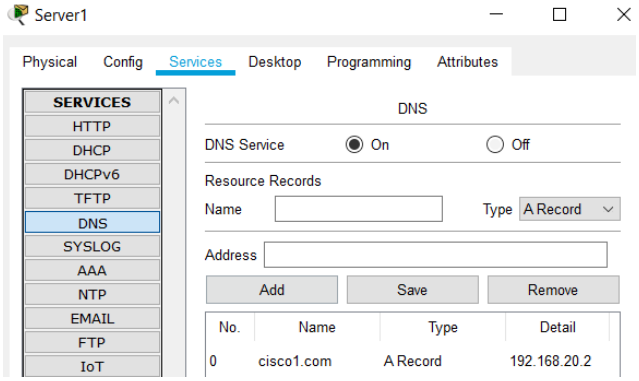


б)

Фиг. 15.14. *Server0 настройки на услуги а) DHCP и б) DNS*



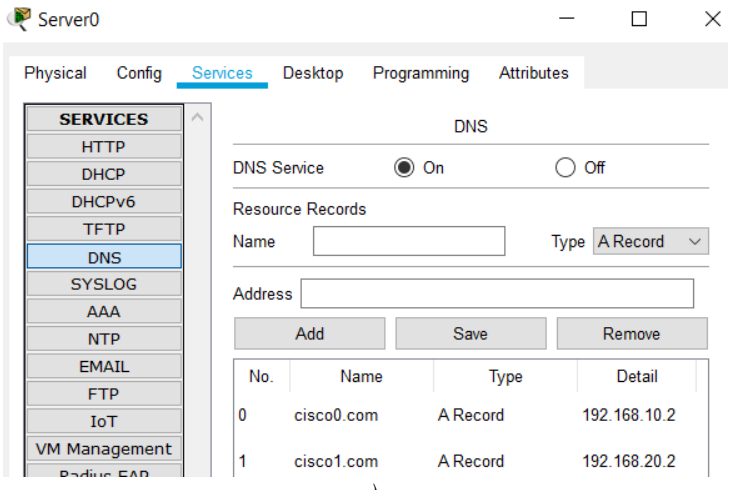
а)



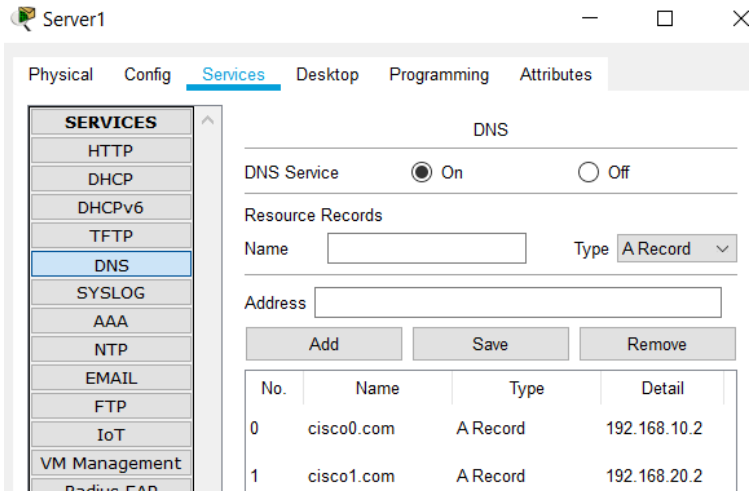
б)

Фиг. 15.15. *Server1 настройки на услуги а) DHCP и б) DNS*

❖ В резултат от предходната стъпка е направено включване на домейн “cisco0.com” за Server0 и “cisco1.com” при Server1. След това следва да се добави всеки от създадените домейни за потребителски достъп до мрежово сървърно звено, за което не е указан, съответно “cisco0.com” при Server1 и “cisco1.com” за Server1. Процесите са онагледни на фиг. 15.16;

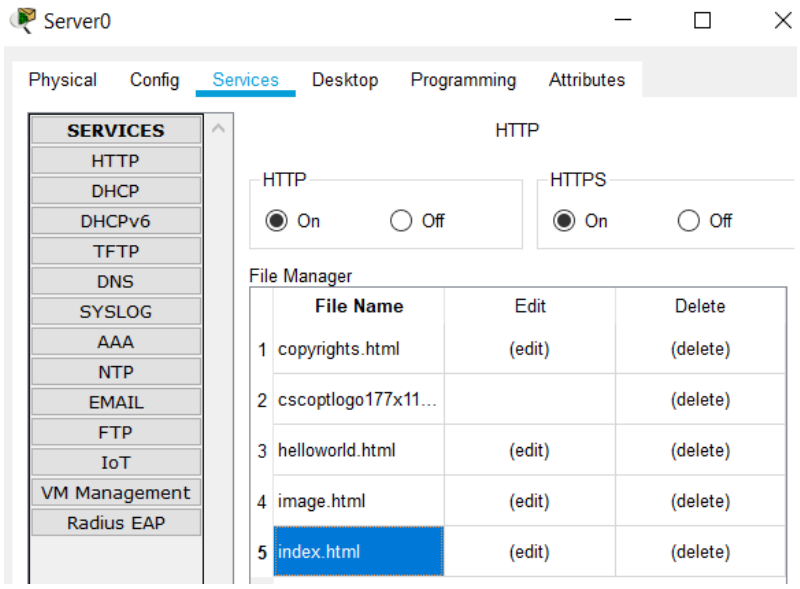


а)

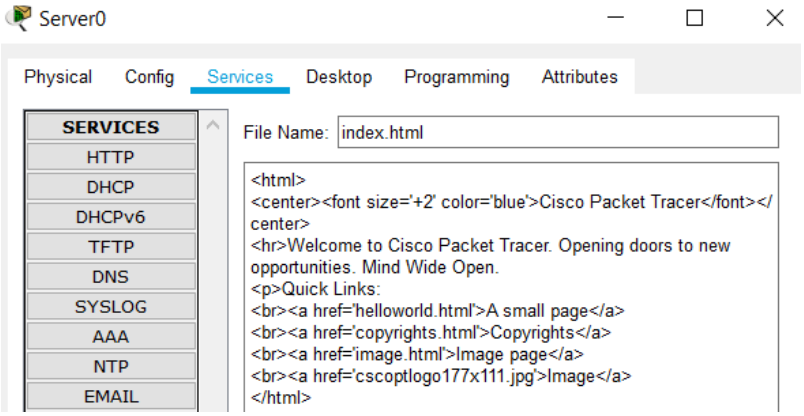


б)

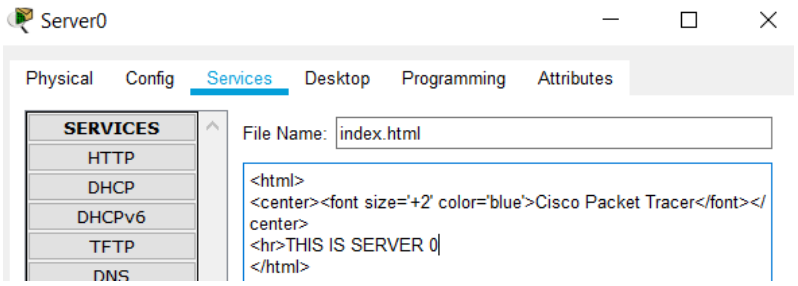
Фиг. 15.16. *Включване на допълнителни домейни към сървърните станции а) "cisco1.com" към Server0 и б) "cisco0.com" към Server1*



а)



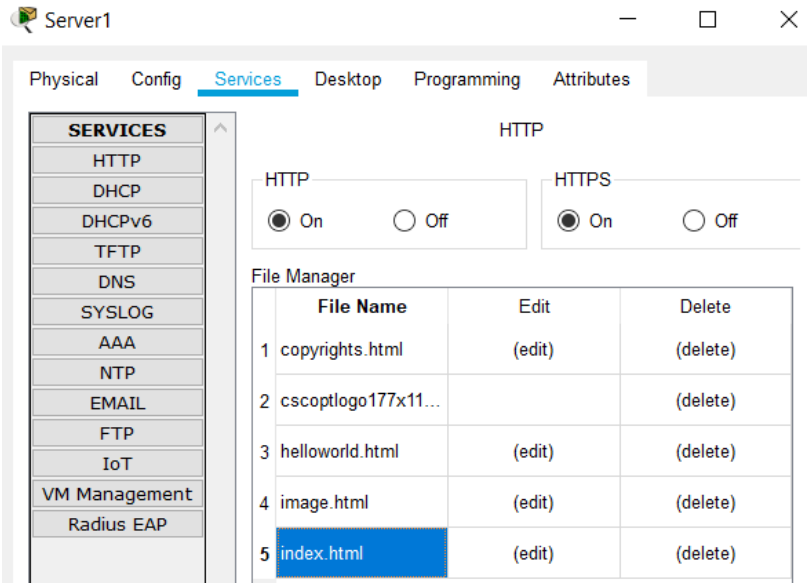
б)



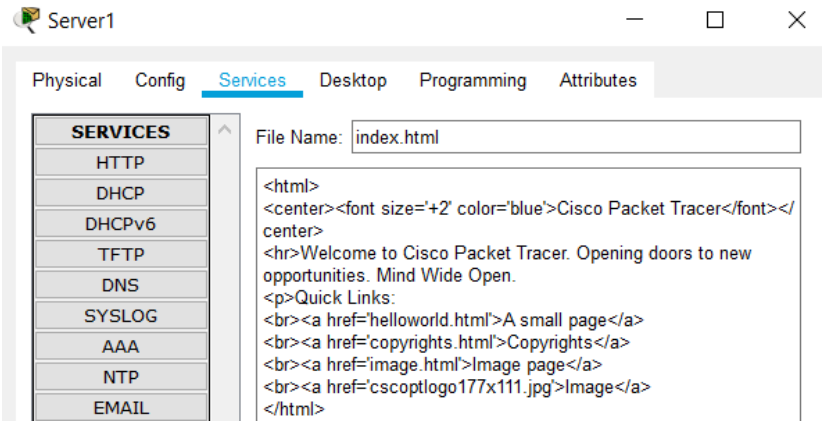
в)

Фиг. 15.17. *Редакция на достъпното WEB съдържание при Server0 - а) "index.html", б) съдържание преди редакция и в) HTML с направени промени*

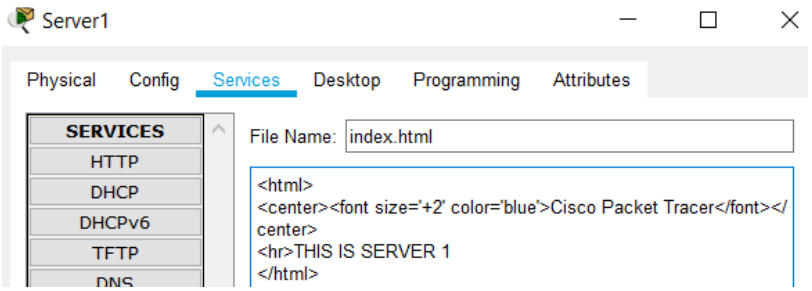
- Редакция на достъпните страници на WEB HTML съдържанието на домейни "cisco0.com" и "cisco1.com". Тук е необходимо да се укаже файл "index.html" и избере "(edit)" при услуга HTML, след което трябва да се зареди съдържанието на документа, направят и съхранят съответните изисквани потребителски промени. В случая се въведат пасажите "THIS IS SERVER 0" и THIS IS SERVER 1". Дейностите са онагдени на фиг. 15.17 и фиг. 15.18;



а)

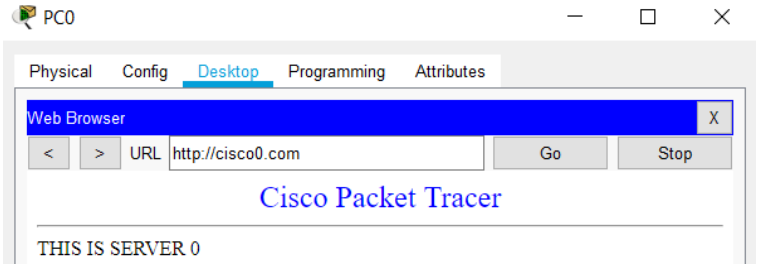


б)

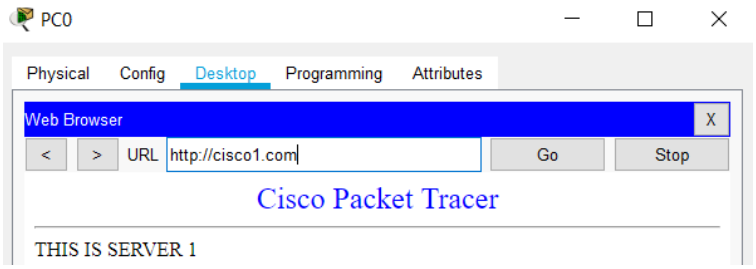


в)

Фиг. 15.18. *Редакция на достъпното WEB съдържание при Server0 - а) "index.html", б) съдържание преди редакция и в) HTML с направени промени*



а)



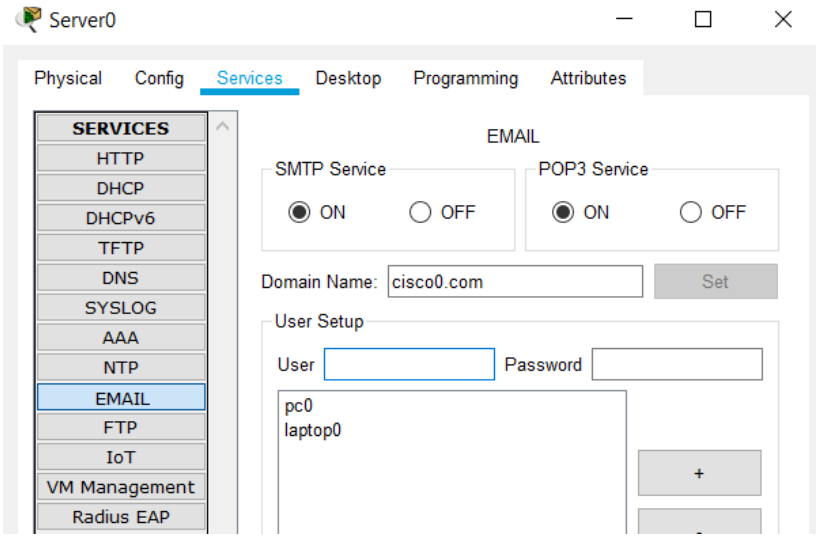
б)

Фиг. 15.19. *WEB достъп от PC0 до съдържанието на домейни а) "cisco0.com" и б) "cisco1.com"*

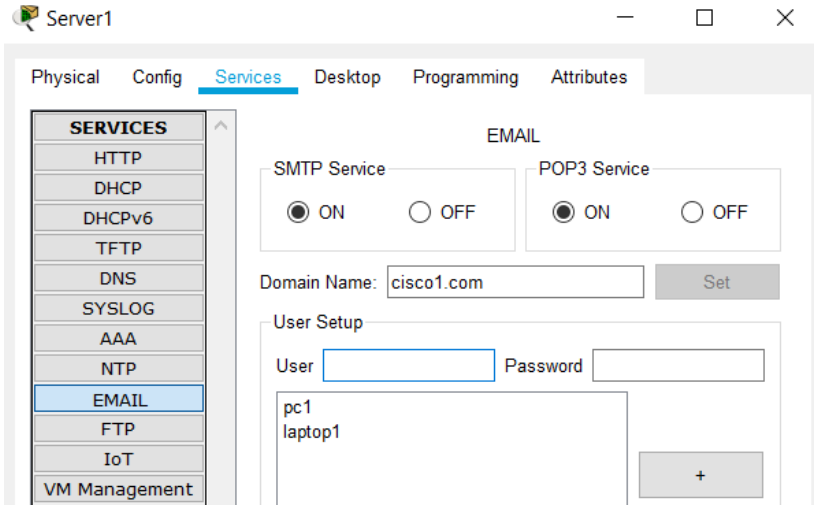
❖ Проверка на достъпа и зареждане на съдържанието на домейни "cisco0.com" и "cisco1.com" през приложението WEB Browser по избор от съответна потребителска компютърна станция. За конкретния случай е изведено достъпното WEB съдържание за мрежови домейни от PC0 на фиг. 15.19 (проверката може да бъде проведена и чрез други персонални станции);

- **<u>Фаза №3: Управление на Email достъпа</u>**

❖ Назначаване на системни потребители относно достъпа до и управление на изпращането и получаването на Email съобщения през домайни "cisco0.com" и "cisco1.com" чрез SMTP (Simple Mail Transfer Protocol) и POP3 (Post Office Protocol версия 3) протоколи при сървърните станции Server0 и Server1 чрез услугата EMAIL, както е показано на фиг. 15.20;



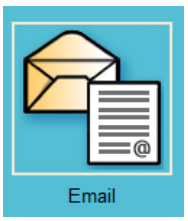
а)



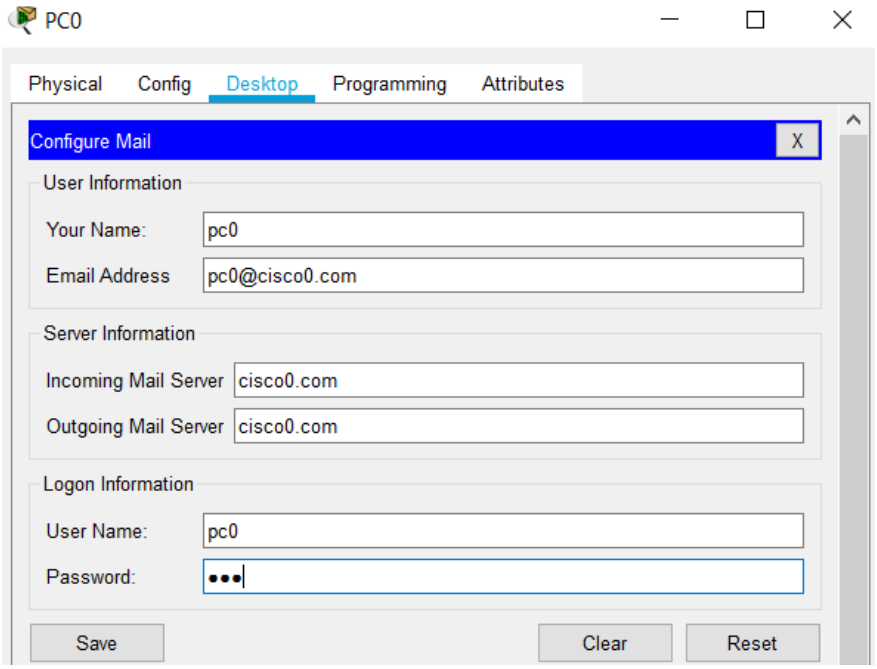
б)

Фиг. 15.20. *Назначаване на системни потребители за Email достъп до домейни а) "cisco0.com" и б) "cisco1.com"*

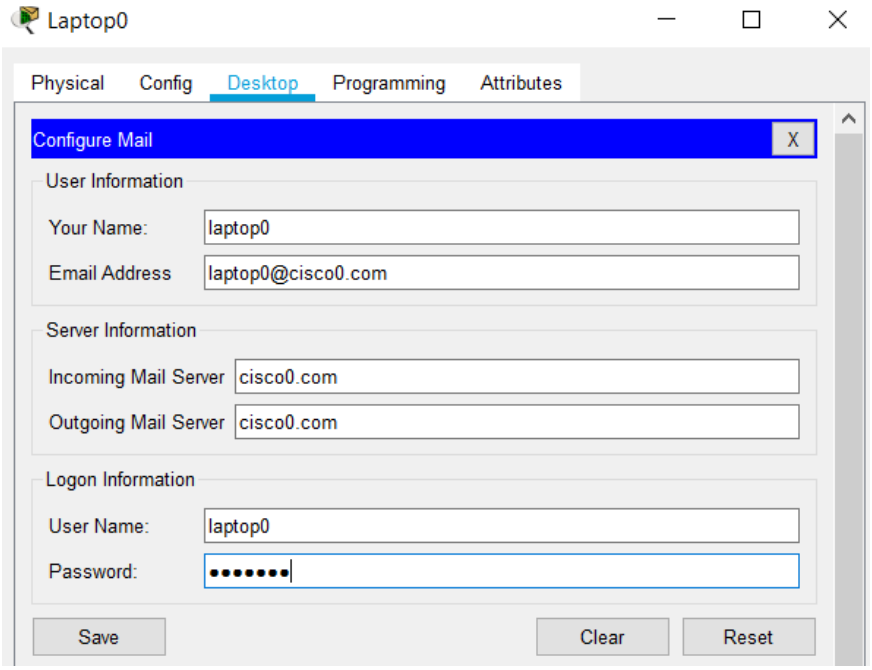
- Създаване на Email потребителски профили по отношение на персоналните компютърни станции PC0, Laptop0, PC1 и Laptop1 във връзка с „потребителско име", „потребителски email адрес", „домейн на входящия трафик" и „домейн за изходящия информационен трафик", както и „име" и „парола на Log on оторизация". Дейностите чрез приложение Email (фиг. 15.21) са последователно изпълнени на фиг. 15.22;



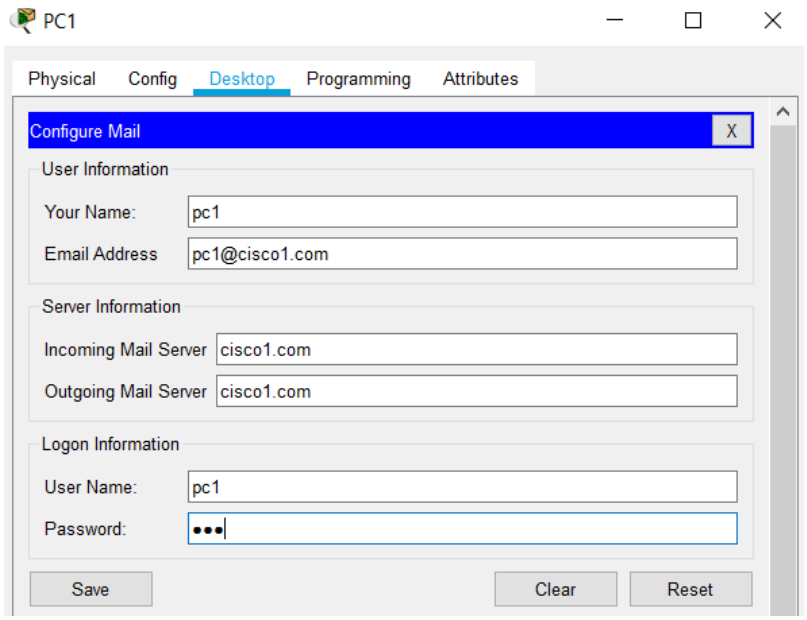
Фиг. 15.21. *Приложение за създаване на потребителски профили за изпращане/получаване на email съобщения*


PC0
Physical
Config
Desktop
Programming
Attributes
Configure Mail
X
User Information
Your Name:
pc0
Email Address
pc0@cisco0.com
Server Information
Incoming Mail Server
cisco0.com
Outgoing Mail Server
cisco0.com
Logon Information
User Name:
pc0
Password:
Save
Clear
Reset

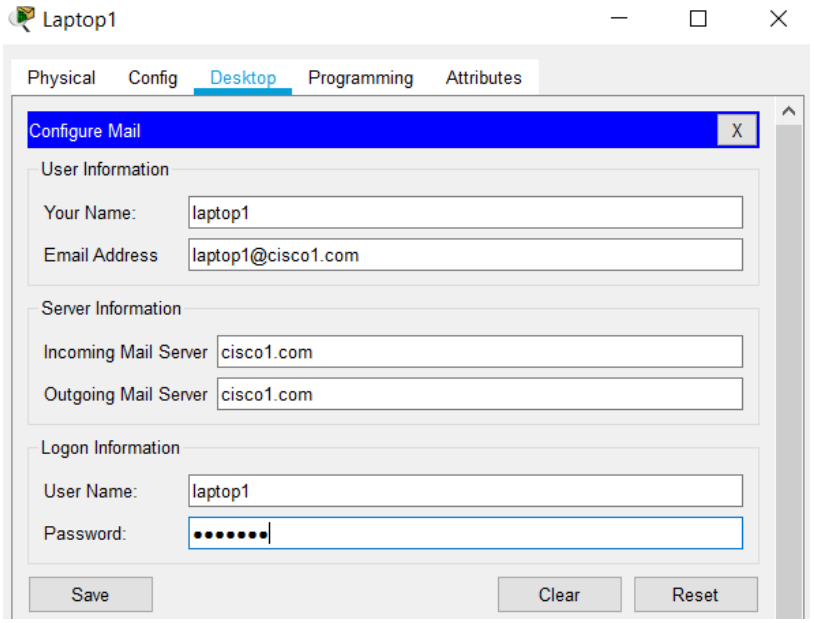
а)


Laptop0
Physical
Config
Desktop
Programming
Attributes
Configure Mail
X
User Information
Your Name:
laptop0
Email Address
laptop0@cisco0.com
Server Information
Incoming Mail Server
cisco0.com
Outgoing Mail Server
cisco0.com
Logon Information
User Name:
laptop0
Password:
Save
Clear
Reset

б)

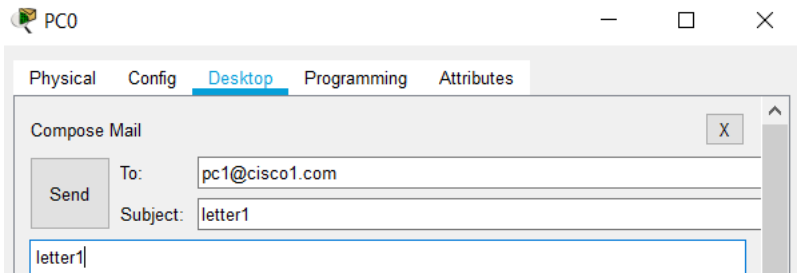


в)

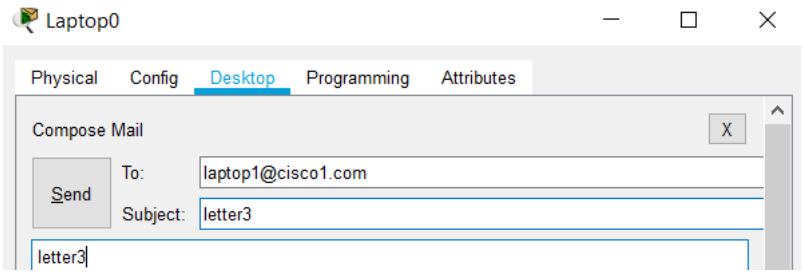


г)

Фиг. 15.22. *Създаване на потребителски профили за изпращане/получаване на email съобщения за персоналните компютърни станции*
а) PC0, б) Laptop0, в) PC1 и г) Laptop1

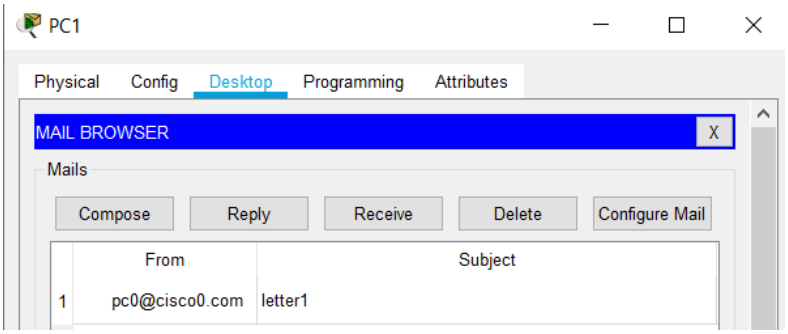


а)

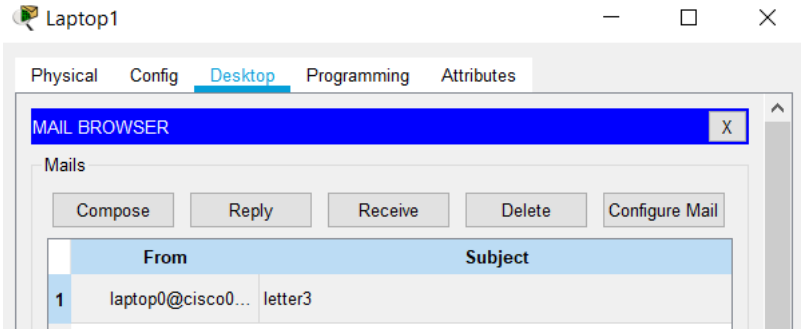


б)

Фиг. 15.23. *Изпращане на email съобщения от а) PC0 към PC1 и б) Laptop0 към Laptop1*



а)



б)

Фиг. 15.24. *Преглед на получените email съобщения при а) PC1 и б) Laptop1*

- Провеждане на тестове чрез изпращане на Email съобщения между създадените потребителските профили от персонални станции между PC0 и PC1, Laptop0 и Laptop1, онагледени на фиг. 15.23 и фиг. 15.24.

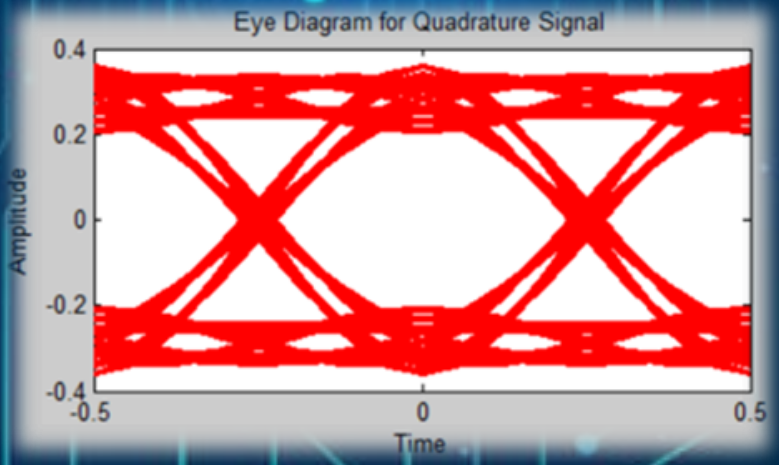
15.4. Контролни въпроси.

1. Кои са основните категории модели „облачни услуги“?
2. По какъв начин се унифицират серийните интерфейси при разглежданата облачна конфигурация във фаза № 1?
3. Какво е процедурното предназначение на Frame Relay стандарта във фаза № 1?
4. С какво се свързва активацията на DHCP услугата в обхвата на сървърните структурни звена във фаза № 2?
5. За какво служи активазацията на настройките на SMTP и POP3 протоколи във фаза № 3 по отношение на тестовите e-mail съобщения?
6. Посочете основните заплахи за сигурността при действия в облачна среда.

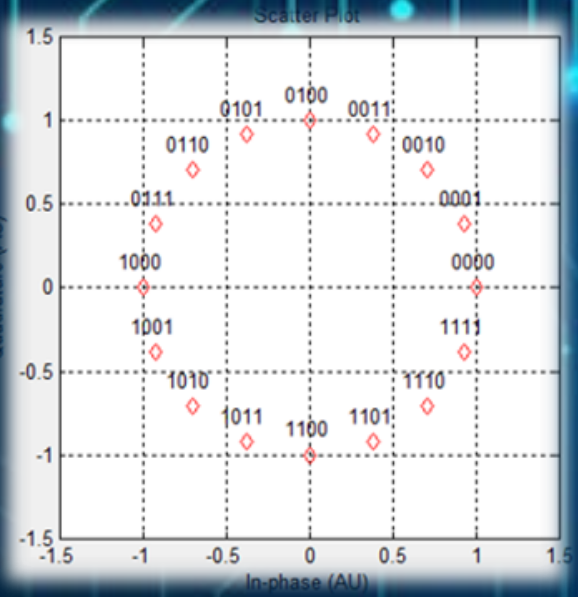
И З П О Л З В А Н А Л И Т Е Р А Т У Р А

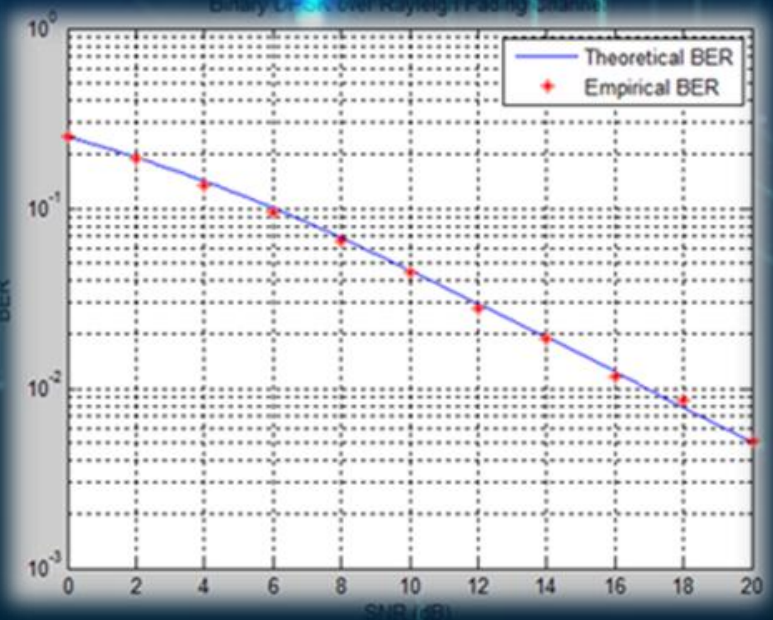
1. S. Haykin, M. Moher, *Communication Systems*. Wiley, 2021.
2. J. Sztrik, *Basic Queueing Theory*. GlobeEdit, OmniScriptum GmbH & Co, 2016.
3. S. Bhooshan, *Fundamentals of Analogue and Digital Communication Systems*. Springer, 2022.
4. Jain N., A. Yadav, A. Jain, *Analog and Digital Communication*. Ashirwad, 2019.
5. D. Hercog, *Communication Protocols: Principles, Methods and Specifications*. Springer, 2020.
6. D. Comer, *Internetworking with TCP/IP: Principles, Protocols and Architecture*. Pearson, 2013.
7. R. Scott, *Computer Networking*. Russell Scott, 2019.
8. L. Peterson, B. Davie, *Computer Networks: A System Approach*. Morgan Kaufmann, 2021.
9. Q. Kiser, *Computer Networking and Cybersecurity*. Independently published, 2020.
10. S. Sankaran, S. Gulasekaran, *Wi-Fi 6: Protocol and Network*. Artech House, 2021.
11. D. Medhi, K. Ramasamy, Network Routing: Algorithms, Protocols, and Architectures. Morgan Kaufmann, 2017.
12. H. Weber, Computer Systems and Networking Guide. Hans Weber, 2019.
13. M. Sibley, *Optical Communications: Components and Systems*. Springer, 2020.
14. G. Agrawal, *Fiber-Optic Communication Systems*. Wiley, 2021.
15. P. Gburzynski, *Modeling Communication Networks and Protocols*. Springer, 2019.
16. P. Tadimety, *OSPF: A Network Routing Protocol*. Apress, 2015.
17. H. Barz, G. Bassett, Multimedia Networks: Protocols, Design and Applications. Wiley, 2016.
18. I. Vidal, I. Soto, A. Banchs, J. Garcia-Reinoso, I. Lozano, G. Camarillo, *Multimedia Networking Technologies, Protocols & Architectures*. Artech House Publishers, 2019.
19. E. Coll, *Ethernet, LANs and VLANs*. Teracom Training Institute, 2023.
20. D. Saha, Z. Mistry, Sh. Patil, *Routing, Switching, Port-Sicherheit und VLAN-Trunking-Protokoll*. Verlag Unser Wissen, 2022.
21. D. Diaw, *VPN: Virtual Private Network*. Djibril Chimère Diaw, 2023.
22. G. Blokdyk, IPsec VPN A Complete Guide. 5STARCooks, 2019.
23. Ch. Bell, *Beginning Sensor Networks with XBee, Raspberry Pi, and Arduino: Sensing the World with Python and MicroPython*. Apress, 2020.
24. H. Rashvand, A. Abedi, *Design Solutions for Wireless Sensor Networks in Extreme Environments*. Artech House Publishers, 2018.
25. R. Allison, J. Johnston, M. Wooster, *Sensors for Fire and Smoke Monitoring*. MDPI, 2021.
26. Ph. Parker, *Fire Sensors and Detectors by Components*. ICON Group International, 2022.
27. D. Hanes, G. Salgueiro, P. Grossetete, R. Barton, J. Henry, *IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things*. Cisco Press, 2017.
28. P. Lea, *IoT and Edge Computing for Architects: Implementing Edge and IoT Systems from Sensors to Clouds with Communication Systems, Analytics and Security*. Packt Publishing, 2020.
29. Ch. Binnie, *Cloud Native Security*. Wiley, 2021.
30. D. Comer, The Cloud Computing Book: The Future of Computing Explained. Chapman and Hall/CRC, 2023.

С Ъ Д Ъ Р Ж А Н И Е



MATLAB





JAVA MODELING TOOL