



دانشگاه جامع
علی-کابردی

موسسه آموزشی پژوهشی صنایع ایران

پایان نامه کارشناسی مهندسی فناوری شبکه های کامپیوتری

عنوان پروژه:

معرفی و پیاده سازی یک پروژه در نرم افزار شبیه ساز 14.5 Opnet

استاد راهنما:

جناب آقای مهندس محسنی

دانشجویان:

سید محمد علی طباطبائی نژادی

حامد محمدی

بهمن ۱۳۹۳

صلى الله عليه وآله

عنوان پروژه:

معرفی و پیاده سازی یک پروژه در نرم افزار شبیه ساز 14.5 Opnet

استاد راهنما:

جناب آقای مهندس محسنی

دانشجویان:

سید محمد علی طباطبائی نژادی

حامد محمدی

نیمسال دوم

تاریخ شروع:

۹۳/۱۱/۱۴

تاریخ پایان:

۹۳/۱۱/۱

تقدیر و تشکر

نخستین سپاس و ستایش از آن خداوندی است که بنده کوچکش را در دریای بیکران اندیشه، قطره ای ساخت تا وسعت آن را از دریچه اندیشه های ناب آموزگارانی بزرگ به تماشا نشیند.

از استاد گرانقدرم جناب آقای مهندس محسنی که زحمت راهنمایی این پروژه را بر عهده داشتند، کمال سپاس را دارم.

فهرست

فصل اول

۱ مقدمه
۲ شبکه و نرم افزارهای شبیه ساز
۲ معرفی شبیه ساز OPNET
۳ مزایای OPNET

فصل دوم

۵ اجزای نرم افزار OPNET
۱۰ نحوه کار با نرم افزار

فصل سوم

۱۱ پیاده سازی یک شبکه داخلی کوچک
۳۵ مروری بر پروتکل های MAC در SDH
۳۸ پیاده سازی شبیه سازی
۵۰ طراحی مدل شبکه WDM
۵۶ تعریف مدل شبکه
۶۲ جزئیات نصب نرم افزار Opnet

مقدمه

در این مقاله ابتدا علت استفاده از نرم افزار های شبیه ساز و مزایای آن را بیان می کنیم، سپس قدری راجع به نرم افزار OPNET توضیح داده و به مقایسه آن با سایر شبیه سازها و علت انتخاب OPNET می پردازیم. در نهایت شبیه سازی و نتایج حاصل از آن را ارائه می دهیم.

امروزه با رشد چشمگیر شبکه های کامپیوتری روش هایی که بتوان توسط آن ها رفتار شبکه ها را آنالیز و بررسی کرد از اهمیت فراوانی برخوردارند. همچنین طراحان شبکه برای آزمایش ایده های نوین خود نیاز مند ابزارهایی جهت شبیه سازی رفتار شبکه ها هستند. شرکت های گوناگون برای تدوین یک پروسه طراحی برای رسیدن به عملکرد بهینه شبکه تلاش کرده و سرمایه گذاری های هنگفتی انجام داده اند. در اکثر روال های طراحی، شبیه سازی شبکه که پس از فاز اول (جمع آوری نیازها) و انجام طراحی انجام می گیرد، یکی از مراحل مهم طراحی محسوب می شود. نرم افزارهای شبیه ساز ابزار مناسبی برای شبیه سازی شبکه ها هستند. به خصوص برای شبکه های امروزی که دارای ساختار با پیچیدگی بالا بوده و استفاده از روش های آنالیز ریاضی در بررسی رفتار آنها عملاً به صورت دقیق امکان پذیر نیست.

شبکه و نرم افزارهای شبیه ساز

شبیه ساز های شبکه را از دید گاههای مختلف می توان تقسیم بندی کرد. یکی از این دیدگاهها برای تقسیم بندی شبیه سازی های شبکه روشی است که برای شبیه سازی به کار می برند . به طوریکه در حالت کلی شبیه سازی به دو روش متفاوت صورت می گیرد . روش اول شبیه سازی به کمک آنالیز دقیق و روش دوم شبیه سازی رویدادهای گسسته است . در روش اول برای شبیه سازی از مدل های ریاضی برای حصول نتایج استفاده می شود و شبکه با مجموعه ای از معادلات ریاضی مدل می شود . عیب عمده این روش شبیه سازی در ساده کردن بیش از حد شبکه در این مدل ها و عدم توانایی در شبیه سازی رفتار پویای شبکه هاست . اما در روش دوم شبیه سازی در سطح پائین (در سطح بسته های موجود در شبکه) انجام شده و از این رو نتایج از دقت بالایی برخوردارند . در این روش نتایج با شبیه سازی رویداد های گسسته به دست می آیند . اما بر خلاف روش اول که شبیه سازی با سرعت بالایی انجام می شود ، در این شبیه سازها نتایج با سرعت کمی حاصل خواهند شد . اما راه حل مناسب ، استفاده از ترکیب این دو روش در شبیه سازی با هدف داشتن عملکرد مورد قبول و حفظ دقت مورد نیاز در موارد بحرانی است.

این شبیه سازها را اصطلاحاً شبیه سازهای مرکب یا دو رگه گویند. از مهمترین شبیه سازهای توان به OPNET و NS-۲ اشاره کرد.

معرفی شبیه ساز OPNET

OPNET که در دانشگاه MIT و در سال ۱۹۸۷ ارائه شد، این امکان را برای کاربر فراهم می کند تا به طراحی و مطالعه شبکه های مخابراتی مختلف ، ادوات شبکه ، و پروتکل های موجود در شبکه بپردازد. OPNET یک شبیه ساز تجاری بوده که برای شبیه سازی پروتکل های رایج در شبکه و مدل های مختلف در شبکه های با رنج بسیار متنوع به کار می رود. به طوریکه در این نرم افزار مدل هایی برای شبیه سازی شبکه هایی از ابعاد کوچک گرفته و در حد یک LAN ساده تا شبکه های ماهواره ای بزرگ وجود دارد . در این نرم افزار از شبیه سازی رویداد های گسسته جهت آنالیز عملکرد و رفتار شبکه ها استفاده میشود . این نرم افزار بر روی سیستم های Windows و Solaris قابل نصب است . همچنین برای اجرای شبیه سازی باید کامپایلر C++ یا ANSI روی سیستم شما نصب شده باشد. پس از نصب کامل این نرم افزار که شامل قسمت های Model Library و

Modeler Documentation است (در انتها نحوه نصب با جزئیات ارائه شده است) و اجرای اولیه ی آن جهت وارد کردن اطلاعات مربوط به License نرم افزار ، می توان کار را با OPNET آغاز کرد.

OPNET در مقایسه با سایر شبیه سازها

بر خلاف NS-۲ که open source می باشد، OPNET نیاز به خرید لایسنس دارد که بسیار گران است. واسط گرافیکی OPNET بسیار ساده و جذاب است و برای برنامه نویسی نیاز به آشنایی با زبان C++ , C دارد. گفته میشود این اولین نرم افزاریست که برای شبیه سازی شبکه به بازار آمد اما مدتی NS-۲ به دلیل open source بودن طرفداران بیشتری پیدا کرد. اما در سالهای اخیر به دلیل رونق شبکه های بی سیم و کارایی بی نظیر OPNET در شبیه سازی شبکه های بی سیم و موبایل (متحرک) دوباره OPNET در صدر قرار گرفت. ضمن اینکه OPNET بسیاری از وظایف را به صورت اتوماتیک انجام می دهد و نیاز به برنامه نویسی هر چیز کوچکی از پایین ترین سطح نیست. از دیگر مزایای OPNET ، سرعت اجرای بسیار بالاست. برنامه ای که در NS-۲ شاید نیاز به یک روز وقت برای اجرا داشته باشد در OPNET در عرض چند ثانیه شبیه سازی میشود. البته OPNET از لحاظ سرعت هنوز نمی تواند از MATLAB بهتر عمل کند. همچنین این نرم افزار قابلیت ارتباط با MATLAB و استفاده از برنامه های MATLAB جهت مدلسازی در پایتترین سطح (سطح پروسس) را دارد.

مزایای OPNET

OPNET همانطور که از نام آن (Optimum Network Performance) مشخص است ، نرم افزاری است که دستیابی به عملکرد بهینه را ممکن می سازد . قابلیت های این نرم افزار در شبیه سازی جنبه های مختلف شبکه، از لایه کاربرد گرفته تا لایه های فیزیکی ، حیرت انگیز است. این نرم افزار به طراحان و محققان اجازه می دهد عملکرد پروتکل ها و تجهیزات را با دقت قابل قبولی از طریق شبیه سازی پیش بینی کنند . تجربه کار با نرم افزار های رقیب مانند NS – ۲ ، ما را متقاعد کرده است که OPNET یک تاز عرصه شبیه سازی شبکه است.

OPNET نرم افزاری قوی است ، ولی تنها برای کاربرانی که با قابلیت های این نرم افزار به خوبی آشنا باشند. متأسفانه قابلیت های زیاد، این نرم افزار را بیش از حد پیچیده جلوه می دهد. البته به صورت کلی شبیه سازی شبکه های کامپیوتری با دشواری های خاصی رو به رو است که مزید بر علت می گردد.

قابلیت های این نرم افزار واقعاً پایان ناپذیر است. کاربردهای مهم OPNET عبارتند از:

۱. آنالیز شبکه‌های فعلی از طریق شبیه‌سازی : این نرم افزار مستقیماً می‌تواند فایل‌های تنظیمات روترهای سازندگان مختلف مانند سیسکو را به عنوان ورودی دریافت کرده و سناریوی فعلی شبکه را شبیه‌سازی کند .
 ۲. طراحی بهینه شبکه : با این نرم افزار می‌توان لینک‌ها را به صورت بهینه انتخاب کرد تا از هزینه‌های اضافی جلوگیری شود.
 ۳. شبیه‌سازی عملکرد پروتکل‌های جدید که در محیط‌های تحقیقاتی به وجود آمده‌اند. ایجاد این پروتکل‌ها بر اساس برنامه‌نویسی ++C و ماشین حالت منتهای FSM می‌باشد .
- با توجه به کتابخانه بزرگ این نرم افزار که شامل انواع روترها، سویچ و سایر تجهیزات سازندگان بزرگی مانند سیسکو، Juniper، Com ۳ و غیره است، جز در مواردی محدود نیازی به استفاده از ویرایشگرهای گره و پروسه نیست و تنها از طریق ویرایشگر اصلی که ویرایشگر شبکه نامیده می‌شود می‌توان شبکه را پیاده‌سازی کرد و نتایج شبیه‌سازی عملکرد آن را به صورت نمودارهای مختلف مشاهده نمود.
- OPNET در ارائه نتایج نیز بسیار کارآمد است. این نرم افزار می‌تواند نتایج شبیه‌سازی را علاوه بر نمایش به صورت گراف‌های متعدد، به صورت صفحه وب نیز ارائه کند.
- قابلیت بسیار جالب OPNET امکان وارد کردن فایل‌های تنظیمات روترها و سویچ‌ها شبکه فعلی در نرم افزار است . فناوری‌های بی‌سیم در دهه اخیر در دنیا گسترش فراوانی یافته‌اند. این فناوریها در شبکه‌های محلی بی‌سیم، لینک‌های نقطه به نقطه و لینک‌های نقطه به چند نقطه مورد استفاده ISP ها، شرکت‌های تجاری و شرکت‌های دولتی قرار می‌گیرند. نرم افزارهای مختلفی را می‌توان برای شبیه‌سازی این شبکه‌ها استفاده کرد. با این وجود بیشتر این نرم افزارها یا توانایی شبیه‌سازی و اعمال ترافیک لایه‌ها بالاتر (TCP/IP) را ندارند و یا از قابلیت‌های لازم برای شبیه‌سازی تأثیر مدولاسیون‌ها، پرتوی آنتن هاو محیط انتقال برخوردار نیستند.

از دیگر ویژگیهای شگفت انگیز OPNET:

۱. دارای یک محیط سه بعدی با قابلیت تنظیم ارتفاع، به عنوان مثال برای قرار دادن آنتن گیرنده یا فرستنده در ارتفاع
۲. امکان طراحی سلسله مراتبی (در سه دسته) برای راحتی و سرعت طراحی

۳. امکان تنظیم شرایط محیطی مانند دما، رطوبت و ... برای مطابقت هر چه بیشتر با دنیای واقعی

۴. امکان تعیین موبیلیتی، سرعت و جهت حرکت و ... برای برخی گره ها

۵. گروه پشتیبانی بسیار قوی و پرسرعت در تمام ساعات شبانه روز

OPNET با استفاده از مدل سلسله مراتبی خود (ویژگی شماره ۲ مذکور در بالا)، قابلیت‌های زیادی در شبیه‌سازی فناوری‌های بی‌سیم به نمایش می‌گذارد. برخی از این قابلیت‌ها عبارتند از :

۱. شبیه سازی مدولاسیون‌های مختلف از طریق نمودار انرژی- خطا.

۲. شبیه سازی پرتوی آنتن

۳. شبیه سازی توان ارسالی

۴. شبیه سازی محیط‌های انتشار از طریق Pipe Line ها

۵. شبیه سازی فناوری‌های Wi-Fi، GPRS، UMTS

کتابخانه این شبیه ساز بسیار کامل بوده و شامل آخرین پروتکل های و مدل های ارائه شده در زمینه شبکه است . همچنین این قابلیت را داراست که پژوهشگران شبکه برای خود اقدام به روزرسانی مدل های موجود در آن کنند.

با توجه به مزایا و ویژگی های بسیاری که OPNET در مقایسه با سایر شبیه سازها دارد از این رو ما این نرم افزار را برای شبیه سازی انتخاب کردیم .

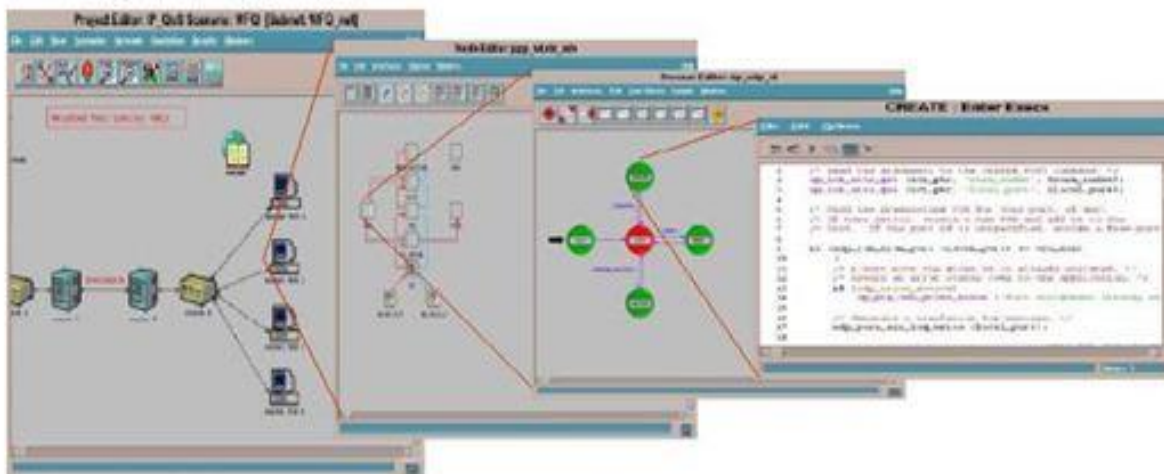
اجزای نرم افزار OPNET

اجزای اصلی OPNET به صورت یک ساختار شامل موارد زیر است:

- Network Model Editor
- Node Model Editor
- Process Model Editor

شبیه ساز OPNET این امکان را فراهم می سازد که مدل شبکه را با ماژول شبکه ایجاد کنیم در هر مدل شبکه نودهای مورد نظر را با ماژول نود طراحی نماییم و رفتار هر گره را با ماژول پروسس تعریف کنیم . ابتدا با ماژول پروسس رفتار گره های مورد نیاز را تعریف کرده و سپس با ماژول

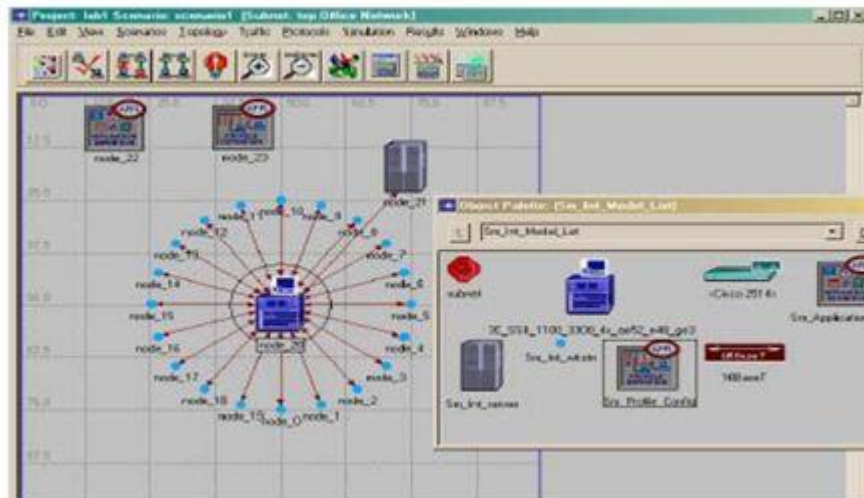
گره ، گره های شبکه را طراحی کرده و سپس آنها را به پروسس تعریف شده پیوند می زنیم . در شکل زیر ساختار ارتباطی بین ماژول های ذکر شده به خوبی قابل مشاهده است:



ساختار ارتباطی بین ماژولها

• ماژول شبکه

هنگامی که در ابتدا یک پروژه را ایجاد می کنیم پنجره پروژه تمام ابزارهای ماژول شبکه را برای مدلسازی شبکه در اختیار کاربر قرار میدهد. این ابزارها به کاربر این امکان را می دهد که ابتدا یک سناریو (مانند شبکه LAN با ابعاد سازمانی و یا یک شبکه WLAN) را انتخاب کند و بر اساس این سناریو ابزارهای و قطعات مناسب را در اختیار گرفته و طبق توپولوژی مورد نظر مدل شبکه را پیاده سازی نماید. مدل شبکه از گره ها ، لینک های ارتباطی و زیرشبکه ها تشکیل می یابد. گره ها می توانند هر قطعه از شبکه مانند سرور، ایستگاه سیار همراه، روترها و غیره باشند. اما زیر شبکه ها می توانند برای گروه بندی استفاده گردند برای نمونه یک شبکه WLAN در دانشکده می تواند شامل اتاق هایی باشد که هر اتاق با یک زیرشبکه مدلسازی شده است و در هر اتاق چندین کامپیوتر سیار بی سیم موجود است. لینک ها امکان اتصال بین گره ها را فراهم می سازند برای مثال یک لینک اترنت چندین کامپیوتر را به یکدیگر متصل می گرداند.

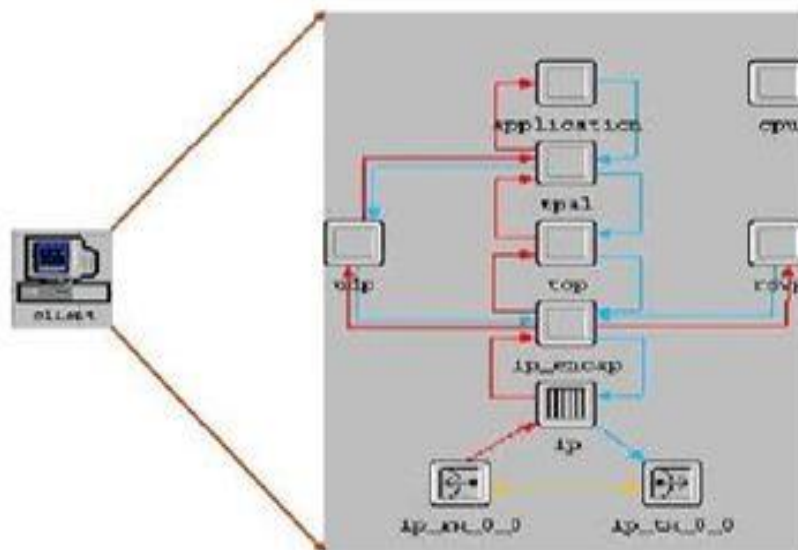


مثالی از ماژول شبکه

• ماژول گره

مدل هر نود شامل مولفه های عملکردی شبکه میشود بعنوان مثال صف (queue) منبع (source)، پردازشگر (processor) و برنامه کاربردی (application) از جمله این مولفه ها می باشند.

همانطور که در شکل زیر مشاهده می شود مدل یک کلاینت شامل مولفه های مختلف شبکه می تواند اجزای یک گره را به تصویر درآورد.



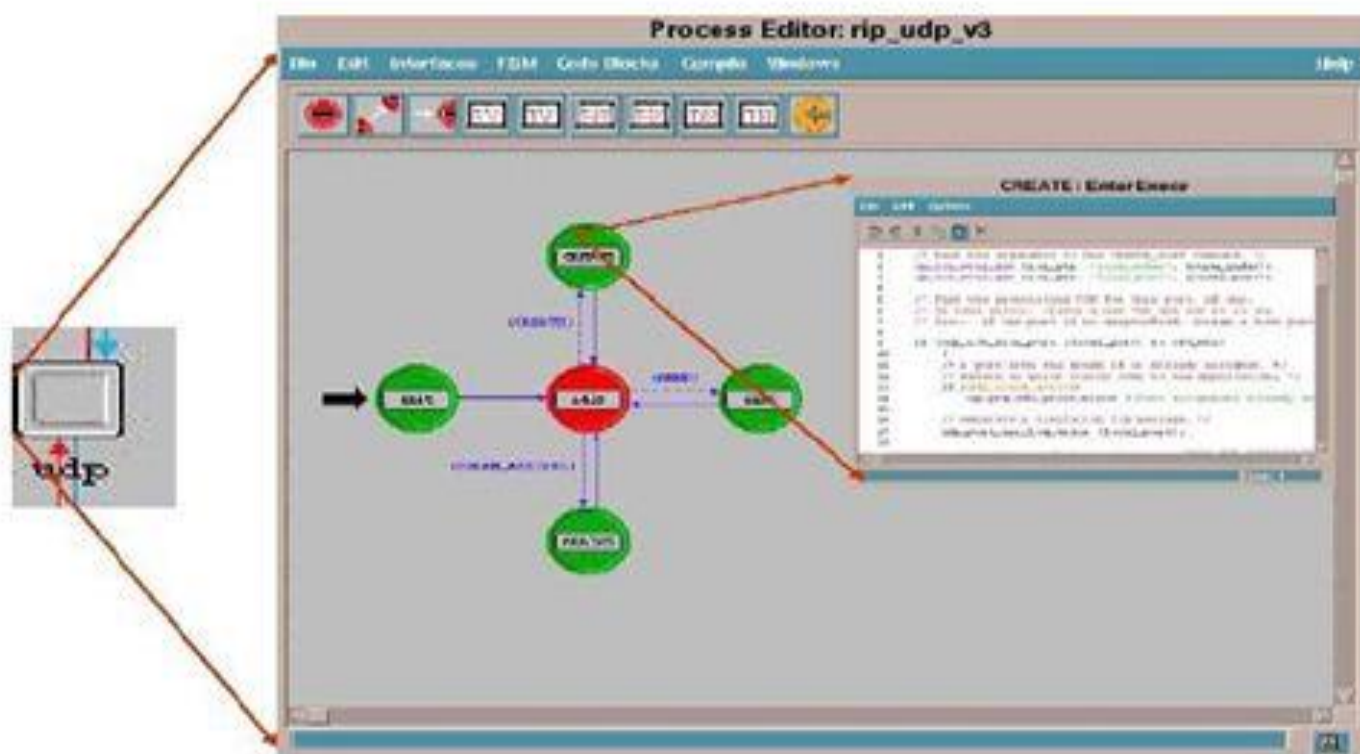
مثالی از ماژول گره

• ماژول پروسس

ماژول پروسس این امکان را به کاربر می دهد که رفتار یک گره را مدلسازی نماید این مدلسازی از طریق نمودار حالت های گذر **diagram transition state**، بلوکهای برنامه به زبان **C** و توابع توکار **Built-in** نرم افزار **OPNET** پیاده سازی میگردد.

یک مدل پروسس می تواند شامل دیاگرام های حالت گذر یا **Finite State Machine (FSM)** و ماژولهای برنامه به زبان **C** و نیز متغیرهای حالت و یا موقتی باشد. هر حالت می تواند به هنگام ورود به آن و یا خروج از آن یک مجموعه کد برنامه را اجرا کند علاوه بر این یک حالت می تواند به دو شکل رفتار نماید یک نوع آن موسوم به **Forced** به هنگام ورود به حالت، کد اجرایی مربوط به ورود را اجرا می کند و بدون درنگ به منظور خروج از حالت، کد اجرایی مربوط به خروج را اجرا می کند و شرایط خروج و گذر به حالت بعد را بررسی می کند در صورت درست بودن شرط به حالت بعدی می رود. در مقابل یک حالت میتواند به شکل **Unforced** عمل نماید به این معنا که پس از اجرای کد

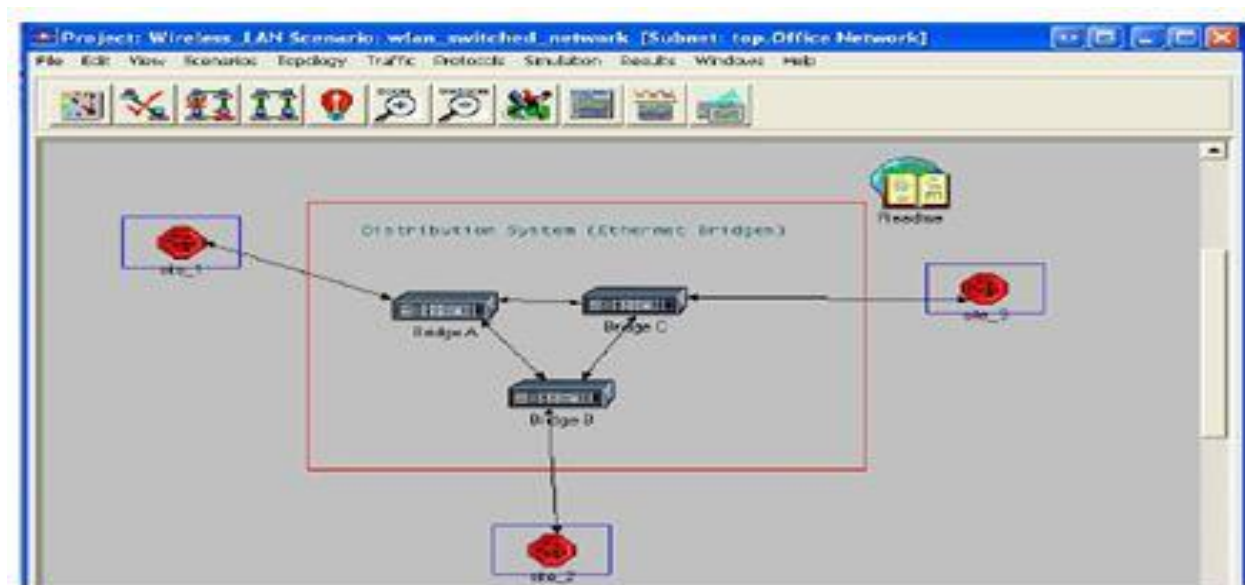
مربوط به ورود آن حالت، کنترل را به هسته برنامه انتقال داده و در آن حالت در انتظار یک سیگنال وقفه باقی می ماند با دریافت وقفه میتواند کد خروج را اجرا نماید.



مثالی از ماژول پروسس

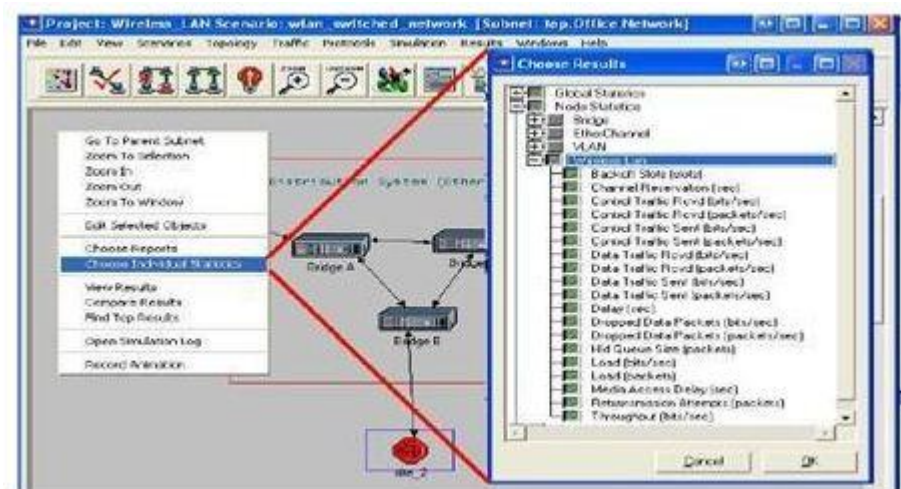
نحوه کار با نرم افزار

ابتدا مدل شبکه ایی که می خواهیم بر روی آن کار نماییم را ایجاد می کنیم در این مدل نوع شبکه، توپولوژی آن و اجزاء تشکیل دهنده شبکه را تعریف می کنیم. در زیرمدل شبکه ای را در پنجره Project مشاهده می کنیم. توجه کنید که در این مرحله هر سه ماژول مذکور بکار گرفته می شود.



مثالی از شبکه LAN بی سیم

پس از ایجاد مدل شبکه، مشخصه های آماری (متغیرهای تحت مشاهده) را که توسط شبیه سازی، تحلیل و به نمایش در خواهند آمد را تعیین می کنیم از جمله متغیرهایی مانند تاخیر، گذردهی سیستم، نرخ بسته های خطا و غیره. برای این منظور از منوی راست کلیک در پنجره حاوی مدل شبکه گزینه "Choose Individual Statistics" را انتخاب می کنیم. همانطور که در زیر مشاهده می شود در کادر Results متغیرهای مورد نظر را می توانیم معین کنیم.



تعیین مشخصات آماری

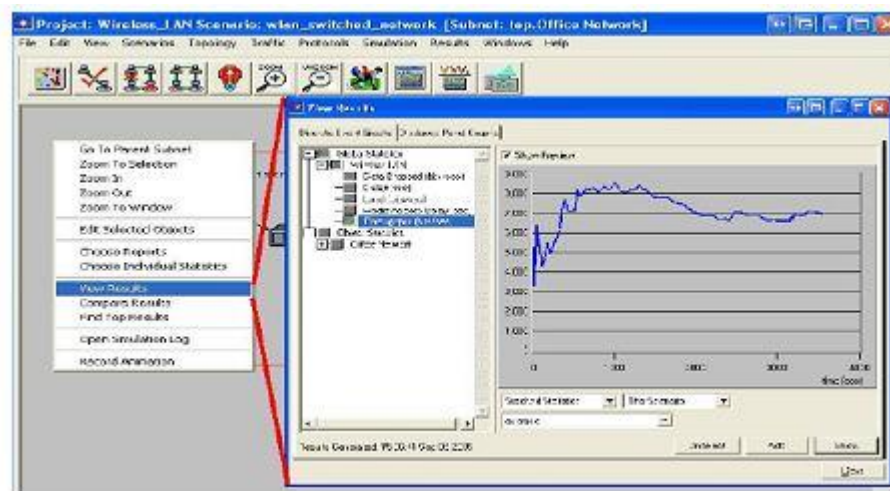
در این مرحله تحلیل مدل شبکه را با اجرای شبیه ساز به انجام میرسانیم برای این منظور کافیت بر روی گزینه **Configure Discrete Event Simulation** کلیک نمائیم این گزینه همانطور که در شکل مشاهده می شود در منوی **Simulation** قرار دارد.



تنظیمات شبیه سازی شبکه

آخرین مرحله در شبیه سازی شبکه ، مشاهده و تحلیل نتایج حاصل میباشد تحلیل به معنای مشاهده تغییرات

حاصل در مشخصه های آماری در اثر تغییر پارامترهای مختلف شبکه میباشد. برای مثال تغییرات گذردهی شبکه به هنگام تغییر در بار ترافیک شبکه.



نتایج شبیه سازی شبکه

پیاده سازی یک شبکه داخلی کوچک

معرفی

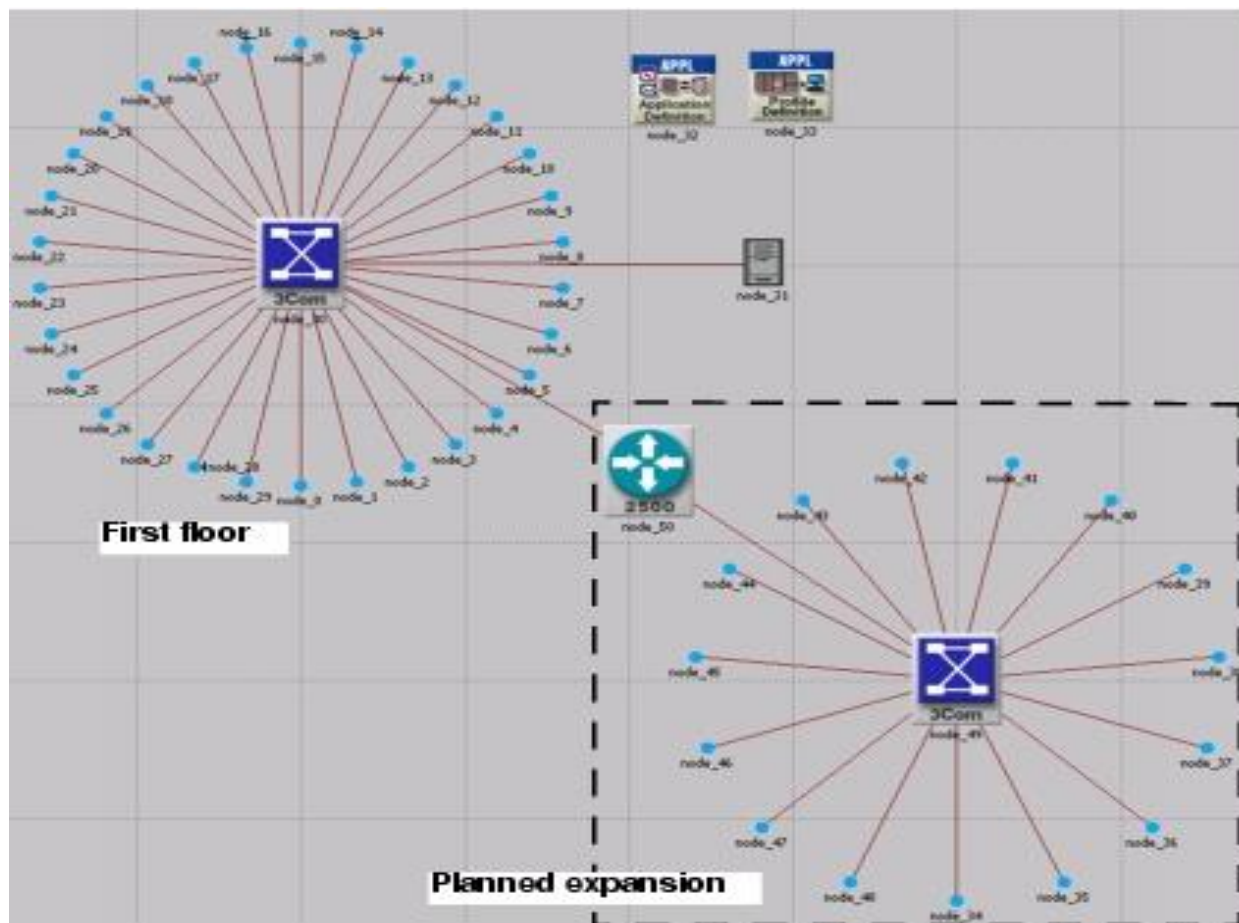
در این مثال شما خواهید دید که چگونه OPNET Modeler یک شبکه را ایجاد و شبیه سازی میکند. عملیات سنجش آن را انجام می دهد. این مثال بر روی نحوه استفاده از Project Editor فوکوس کرده و چگونگی استفاده با نود و پروسس را آموزش خواهد داد. همچنین در این مثال بر روی موارد زیر را می آزمائیم

- طراحی و پیاده سازی سریع یک شبکه
- جمع آوری آمار در مورد عملکرد شبکه
- تجزیه و تحلیل این آمار

در این مثال ، از Project Editor برای ساخت یک توپولوژی جهت یک شبکه داخلی کوچک بهره برده و اطلاعات آماری جهت اجرای شبیه سازی و تجزیه و تحلیل نتایج جمع آوری میشود .

در این مثال ، شما عملیات گسترش اینترنت کوچک یک شرکت را برنامه ریزی می کنید . در حال حاضر، این شرکت دارای یک شبکه با توپولوژی ستاره در طبقه اول ساختمان اداری و برنامه های خود را برای اضافه کردن یک شبکه با توپولوژی ستاره در یکی دیگر از طبقات می باشید . شما این سناریو را تا جایی که شبکه دوم منجر به خطا در شبکه اول نشود ادامه خواهید داد .

شبکه نهایی بصورت زیر خواهد بود



پیاده سازی

در هنگام ایجاد مدل شبکه جدید، برای بار اول شما باید **project** و **scenario** جدید ایجاد کنید. یک پروژه یک گروه از سناریوهای مرتبط که هر کدام تشریح کننده جنبه های مختلف شبکه است. پروژه ها می توانند سناریوهای متعدد باشد.

پس از ایجاد یک پروژه جدید می کنید، شما با استفاده از یک Wizard میتوانید سناریو جدید را ایجاد کنید. گزینه ای زیر در این Wizard به شما اجازه میدهد

- تعریف توپولوژی اولیه از شبکه
 - تعریف مقیاس و اندازه شبکه
 - انتخاب نقشه پس زمینه برای شبکه
 - تعریف و انتخاب object palette جهت استفاده در پروژه
- این ویزارد به ازای هر بار اجرای دستور **New Project** در اختیار شما قرار میگیرد .
- برای استفاده از **Setup Wizard** جهت ایجاد یک سناریوی جدید مراحل زیر را انجام دهید:
۱. در صفحه اصلی **Opnet Modeler** از منوی **File** گزینه **New** را انتخاب نمایید
 ۲. گزینه **Project** را از منوی انتخاب و دکمه **OK** را می فشاریم
 ۳. پروژه و سناریو را مطابق با الگوی زیر نام گذاری کنید
- نام پروژه **Sm_Int<initials>** که **Initials** کلمه ای است که جهت عدم تکراری بودن نام فایل بکار میرود
 - نام سناریو که آن را **First_floor** قرار دهید
- Wizard** باز می شود.
۴. در **Wizard** مقادیر زیر را به عنوان پارامتر انتخاب کنید

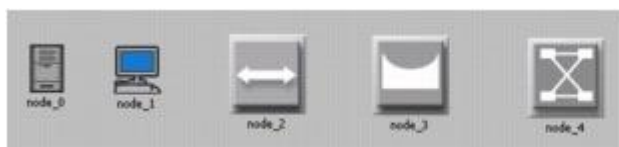
Dialog Box Name	Value
۱. Initial Topology	Select the default value: Create empty scenario.
۲. Choose Network Scale	Select Office . Select the Use metric units checkbox.
۳. Specify Size	Select the default size: ۱۰۰ m x ۱۰۰ m
۴. Select Technologies	Include the Sm_Int_Model_List model family.
۵. Review	Check values, then click Finish .

محیط کاری با اندازه انتخابی شما نمایان میشود . پالت اشیا که شما انتخاب نمودید در پنجره جداگانه باز میشود.

ایجاد شبکه

مدل شبکه توسط تودها و لینکها از پالت اشیا در Project Editor ایجاد میگردد

Node : نمایانگر یک شیء واقعی که توانایی ارسال و دریافت داده را دارد



نودها

Link : بستر ارتباطی که نودها را به هم متصل میکند . لینکها نمایانگر اتصالات فیزیکی هستند (الکتریکی یا کابلهای نوری)



لینک

این ابزارها در پالت اشیا قرار دارند . در این پالت کلیه اشیا بصورت گرافیکی در دسترس قرار دارند

به یکی از سه روش زیر ویا بصورت ترکیبی میتوان یک توپولوژی شبکه ایجاد کرد .

۱. روش فراخوانی توپولوژی

۲. روش چیدن اشیا از پالت اشیا در صفحه کاری

۳. روش استفاده از **Rapid Configuration**

Rapid Configuration ایجاد یک شبکه در یک مرحله پس از انتخاب تنظیمات شبکه، انواع گره های

درون شبکه، و انواع لینک هایی که برای اتصال گره بکار میرود.

برای ایجاد شبکه طبقه اول از متد **Rapid Configuration** استفاده کنید

برای ایجاد توسط این روش مطابق مراحل زیر اقدام کنید

۱. از منوی **Topology > Rapid Configuration** را انتخاب نمائید

۲. از منوی کرکره ای کوچک نوع **Star** را انتخاب نموده و **Next** را بفشارید



۳. مقدار **Center Node Model** را به **۳C_SSII_۱۱۰۰_۳۳۰۰_۴s_ae۵۲_e۴۸_ge۳**

تغییر دهید

۴. گزینه **Periphery Node Model** را به **Sm_Int_wkstn** تغییر داده و تعداد را نیز به

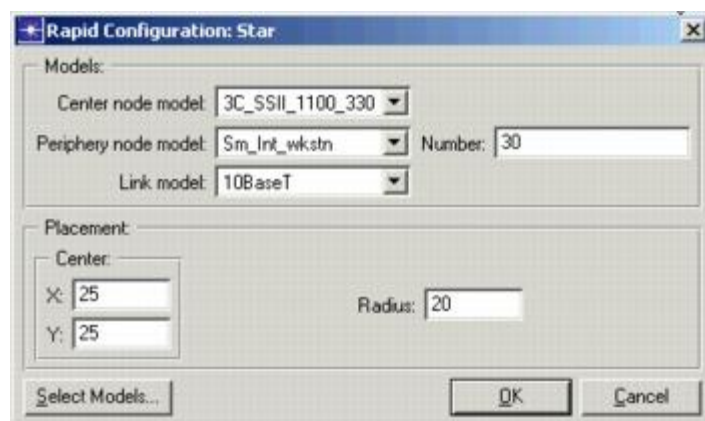
۳۰ تغییر دهید

۵. گزینه **Link Model** را به **BaseT ۱۰** تغییر دهید

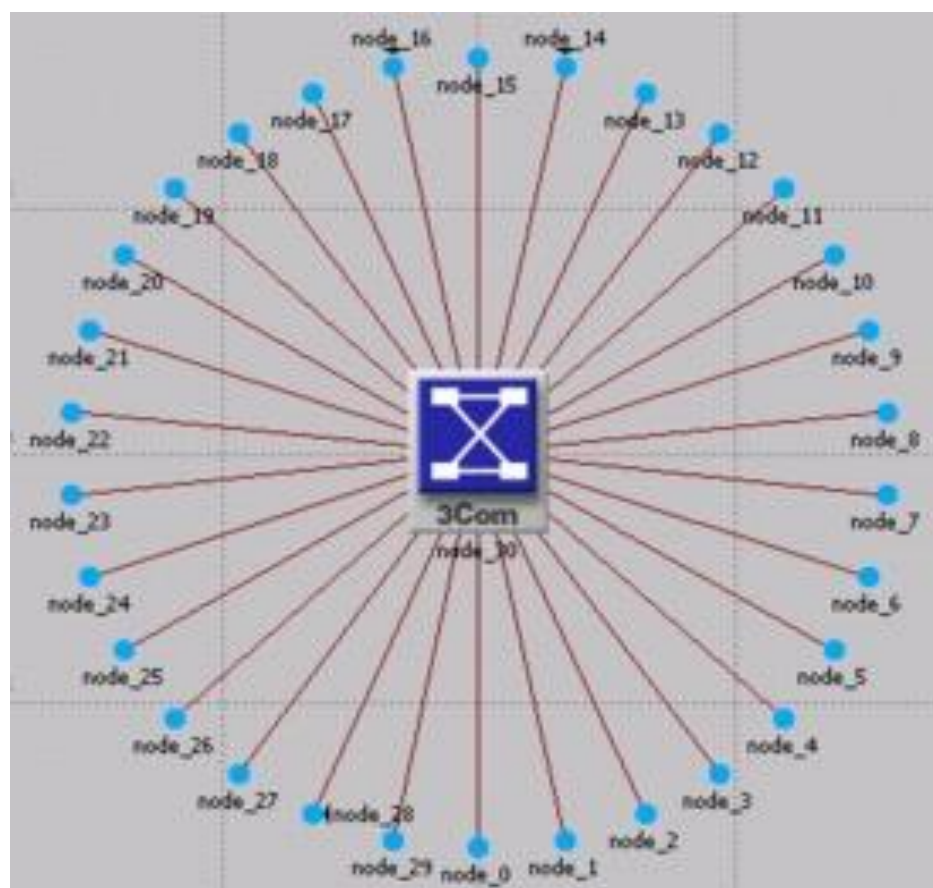
۶. مختصات **X** و **Y** قرار گیری شبکه را انتخاب نمائید

۷. شعاع شبکه را به **۲۰** تغییر دهید

۸. **OK** را بفشارید



شبکه در Project Editor ترسیم میشود



حالا نیاز به ایجاد یک سرور داریم که از پالت اشیا آنرا توسط موس به محیط کار اضافه می نمائیم

۱. اگر پالت شبکه بسته شده است توسط این آیکون آنرا فراخوانی کنید



۲. شیء Sm_Int_server را پیدا کنید و آنرا به وسط صفحه کار بکشید

برای اتصال سرور به شبکه مطابق روال زیر اقدام کنید

۱. شیء BaseT ۱۰ را از پالت اشیا پیدا کرده و بروی آن دابل کلیک کنید

۲. موس را ابتدا بروی سرور فشرده و پس از آن نقطه انتهایی کابل را به سوئیچ متصل نمائید

۳. Right click نمائید تا انتخاب کابل غیر فعال گردد

در انتها باید اشیا را پیکربندی نمائید تا ترافیک نرم افزارها که بروی شبکه موجود هست را مشخص نمائید . پیکربندی و تنظیم مشخصات دستگاهها کار پیچیده ای است که در این مثال از آن صرف نظر می نمائیم . برای این منظور ما از یک پروفایل آماده و یک پیکربندی آماده از درون پالت اشیا استفاده میکنیم که شامل آیتم های زیر است

- یک شی نرم افزاری با تعریف و توزیع و پیکربندی استاندارد
- یک دیتا بیس

برای این منظور مطابق روال زیر اشیا مورد نظر را بروی صفحه کار بکشید

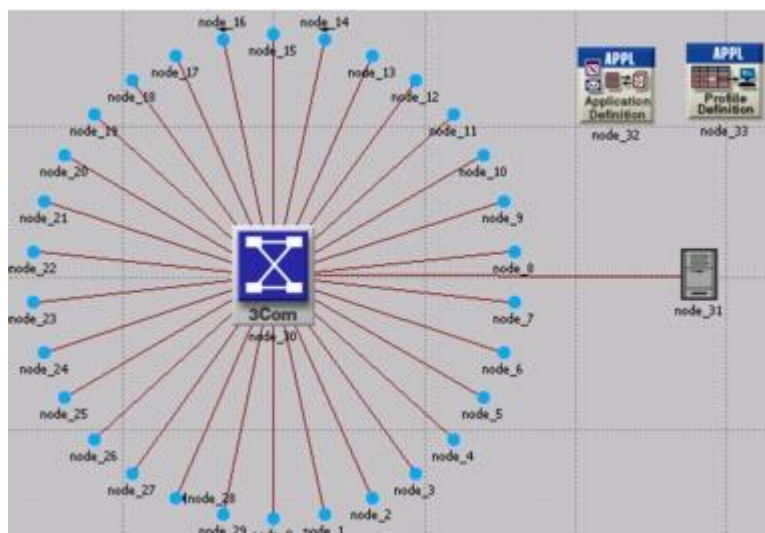
۱. شیء Sm_Application_Config را یافته و بروی صفحه کار بکشید.

۲. Right Click

۳. شیء Sm_Profile_Config را یافته و بروی صفحه کار بکشید .

۴. Right Click

حال شبکه ما می بایست مطابق شکل زیر باشد



شکل نهایی شبکه ما

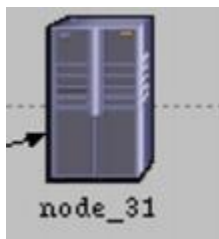
حالا میتوانید به جمع آوری داده بپردازید

هرچند قبل از آن نگاهی خواهیم داشت به ویرایشگر نود و پروسس

ویرایشگر نود و پروسس زیرساختی از جریان کار OPNET Modeler هستند .. ویرایشگر نود جهت ایجاد نود که جریان داخلی داده را درون شبکه تشریح میکند بکار برده میشود. ویرایشگر پروسس که جهت تشریح رفتار منطقی مدل و همچنین نود بکار برده میشود

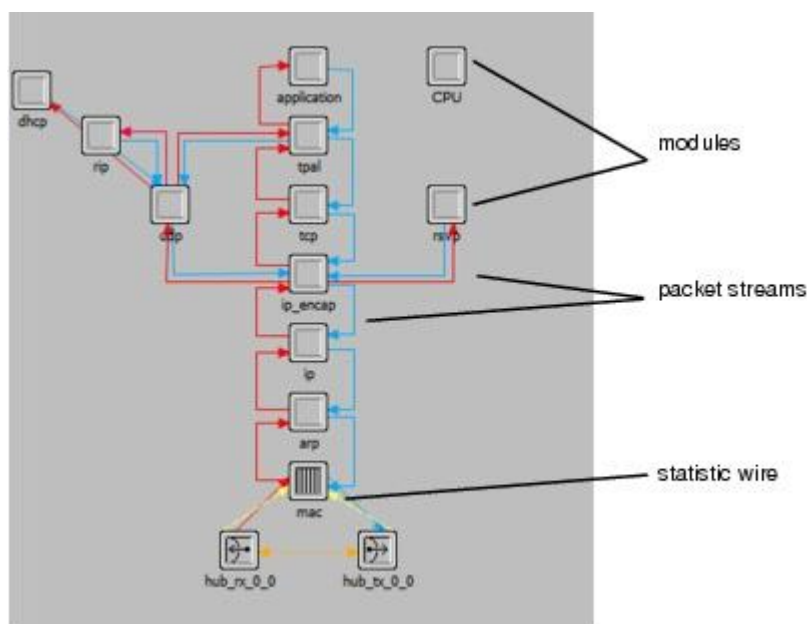
حالا اجازه بدهید نگاهی بیندازیم به گره که سرور را در طبقه اول کنترل میکند

بروی سرور در Object Editor دابل کلیک کنید



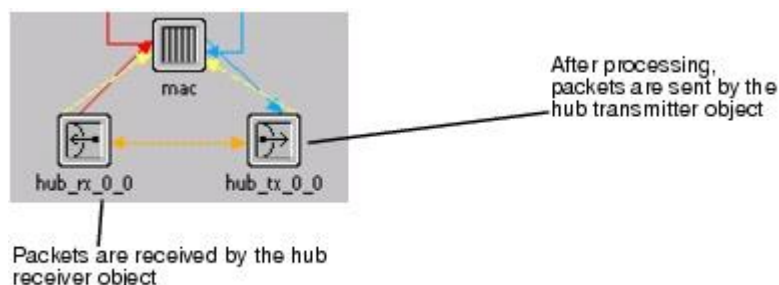
Node Editor در یک پنجره جدید نمایان میشود

این شکل نمایانگر مدل گره با ماژول شبکه می باشد . این نو از چندین ماژول درست شده . Packet Stream و Statistic Wires این ماژولها را به هم متصل کرده است .



مدل گره سرور

در حین شبیه سازی ، Packet ها از ماشین کلاینت ارسال شده و به هاب میرسند . و از پشته پروتکل ها پردازش شده تا به ماژول برنامه کاربردی می رسند . پس از پردازش ، Packet ها مجدداً از پشته پایین فرستاده میشوند تا به هاب رسیده و سپس به ماشین کلاینت بازگردند .

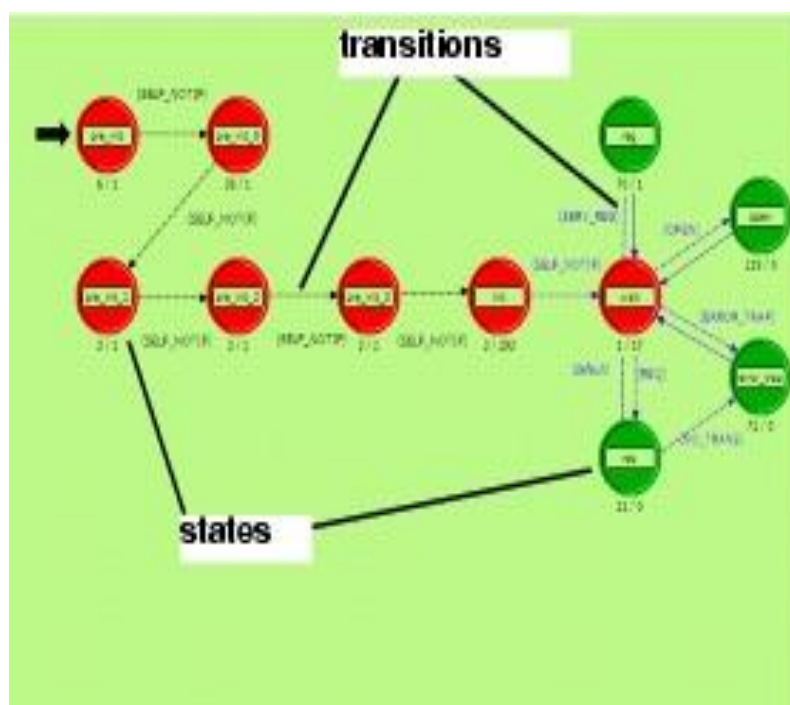


Packet پردازش شده توسط مدل گره

حالا نگاهی بیندازیم به مدل پراسس که رفتار ماژول **tpal** را نمایش میدهد . برای دیدن این مدل پراسس بروی **tpal** دابل کلیک کنید



مدل پراسس در پنجره جدید نمایش داده میشود

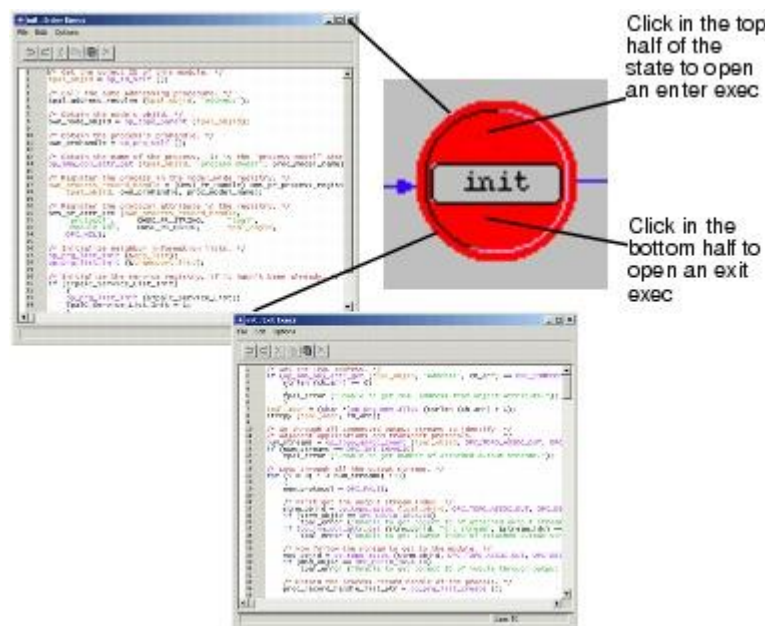


نمایی از پراسس مدل

۱. به ایستگاههای قرمز و سبز دقت کنید و همچنین به خطوط پیوسته و نقطه چین که نمایانگر انتقال بین ایستگاهها و وضعیتها هستند
۲. هر وضعیت در مدل پراسس شامل یک **enter executive** و یک **exit executive** هستند .
اولی زمانی که اجرا می شود که یک پروسه به یک وضعیت برسد . و دومی زمانی اجرا میشود که پروسه به از یک وضعیت خارج شود . این وضعیتها توسط زبان سی گسترش داده شده اند
۳. یک **enter exec** را با دابل کلیک بروی نیمه بالایی وضعیت **init** باز کنید

۴. یک `exit exec` را با دابل کلیک بروی نیمه پایینی وضعیت باز کنید

باز کردن `Enter Exec` یا `Exit Exec` از یک وضعیت



۵. هر دو پنجره اجرا ببندید

جمع آوری داده ها

جمع آوری داده ها به دو طریق `Object statistic` و `Global Statistic` انجام می گیرد

حال که شبکه ایجاد شد باید تصمیم بگیرید که چه اطلاعاتی را باید جمع آوری کنید

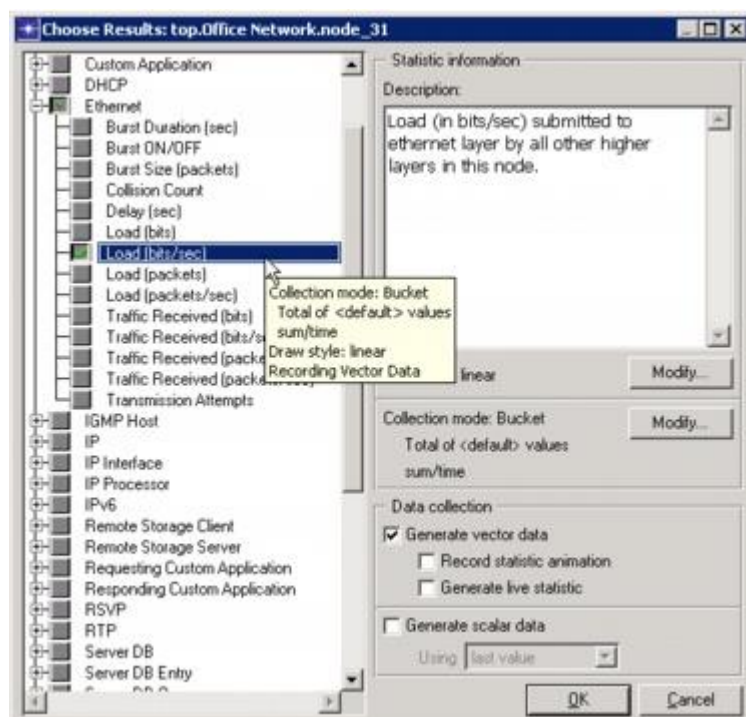
- آیا سرور قادر است تا لود شبکه دوم را نیز هندل کند
- آیا جمع تاخیر (`Ethernet Delay`) در شبکه قابل پذیرش است پس از آنکه شبکه دوم نیز راه اندازی گردید

برای پاسخ به این سوالها باید Performance فعلی شبکه را بسنجید . برای این منظور بایستی اطلاعات آماری شیء را محاسبه نمائید

۱. Server Load
۲. و یک Global Statistic یعنی Ethernet Delay

میزان لود سرور یک اطلاعات کلیدی است که بازتاب Performance تمام شبکه است . بدین منظور روال زیر را انجام می دهیم

۱. بروی سرور Right Click کرده و Choose Individual DES Statistics را از منو انتخاب میکنیم
۲. پنجره نمایش نتیجه باز میشود
۳. برای جمع آوری بار ترافیکی Ethernet
۴. ساختار درختی Ethernet را باز کنید

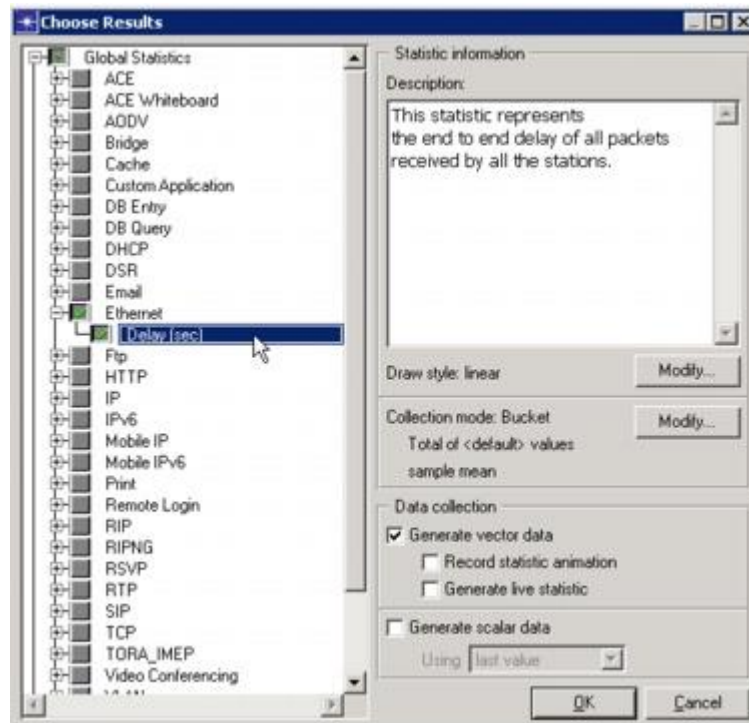


۱. گزینه Load (bits/sec) را انتخاب نمائید

۲. گزینه OK را فشرده تا دیالوگ بسته شود

Global statistics را میتوان برای گردآوری اطلاعات در مورد کل شبکه بکار برد. برای مثال شما میتوانید تاخیر را برای تمام شبکه محاسبه نمایید. برای این منظور

۱. بروی محیط کار Right click کرده و Choose Individual DES را انتخاب نمایید



۲. ساختار درختی Global Statistics را باز کنید

۳. ساختار درختی Ethernet را باز کنید

۴. Delay (sec) را انتخاب نمایید

۵. OK را فشرده تا این پنجره بسته شود

حال اولین شبیه سازی را انجام خواهیم داد

در ابتدا شما می بایست چک کنید که آیا Network Simulation Repositories مقدار appropriately را دارد؟

بدین منظور

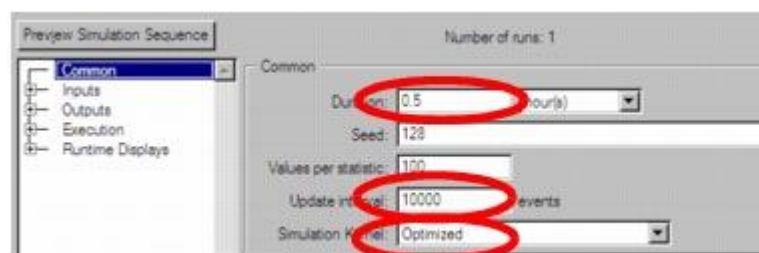
۱. به منوی **Edit > Preferences**. را انتخاب کنید
۲. در قسمت **Search for** عبارت **network simulation** را وارد کرده و دکمه **Search** را بفشارید
۳. اگر مقدار این کلید **stmod** نبود بروی آن کلیک کنید
۴. در پنجره جدید کلید **insert** را فشرده و کلمه **stmod** را در فیلد وارد کنید
۵. دکمه **ok** را دو بار فشرده تا پنجره ها بسته شوند

برای اجرای سیمولیشن

۱. به مسیر **DES > Configure/Run Discrete Event Simulation...** بروید
۲. شما همچنین میتوانید از این دکمه استفاده کنید

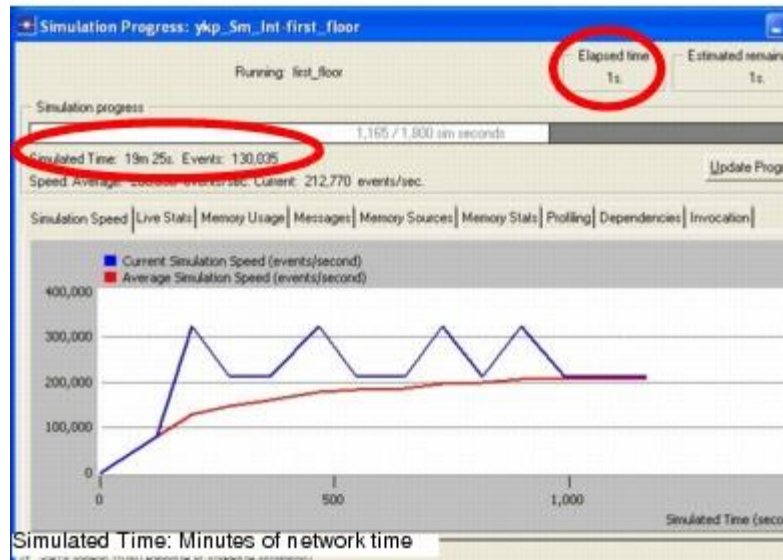


۳. دکمه **Detailed** را بزنید



۴. مقدار ۰/۵ را در قسمت **Duaration** وارد کنید
۵. مقدار ۱۰۰۰۰ رویداد را در قسمت **Update interval** وارد کنید
۶. مقدار **Simulation Kernel** را به **Optimized** تغییر دهید
۷. دکمه **Run** را انتخاب کنید

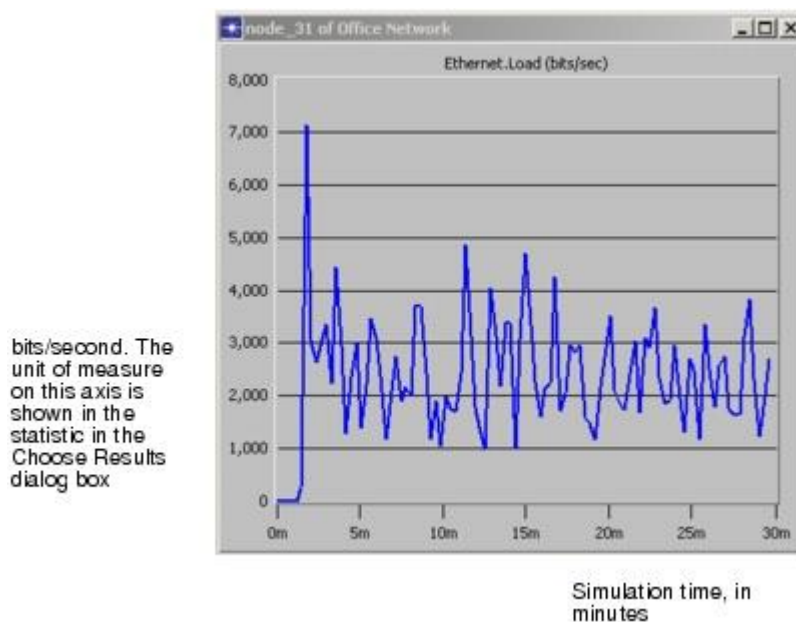
Elapsed Time: Number of seconds the simulation has run



مشاهده نتایج

برای مشاهده نتیجه از روش زیر میتوان استفاده کرد

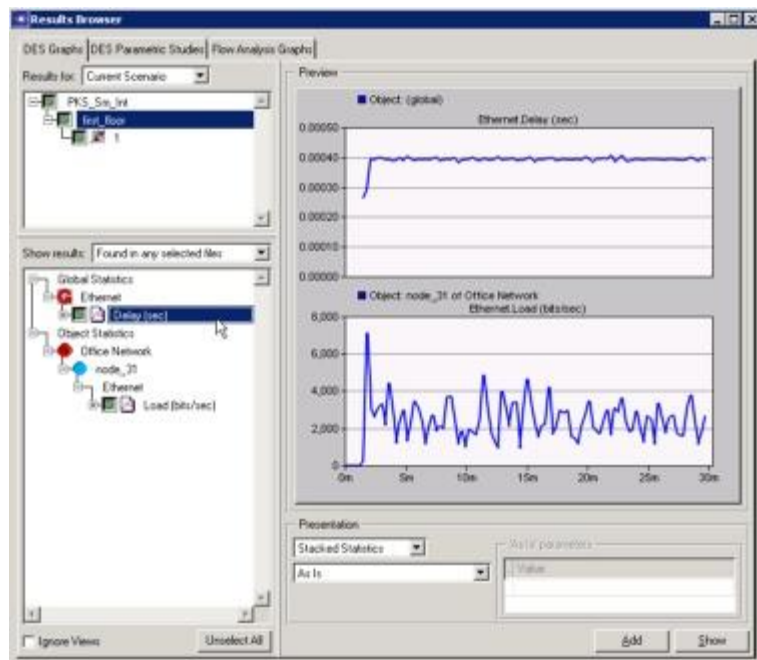
۱. بروی نود مورد **Right Click** کرده و گزینه **view results** را انتخاب می نمایم
 ۲. ساختار درختی **Office network.node_۳۱ > Ethernet** را باز نمائید
 ۳. بروی چک باکس **Load (bits/sec)** کلیک کرده تا انتخاب شود
 ۴. دکمه **Show** را جهت نمایش فشار دهید .
- نتیجه مطابق با شکل زیر نمایش داده خواهد شد



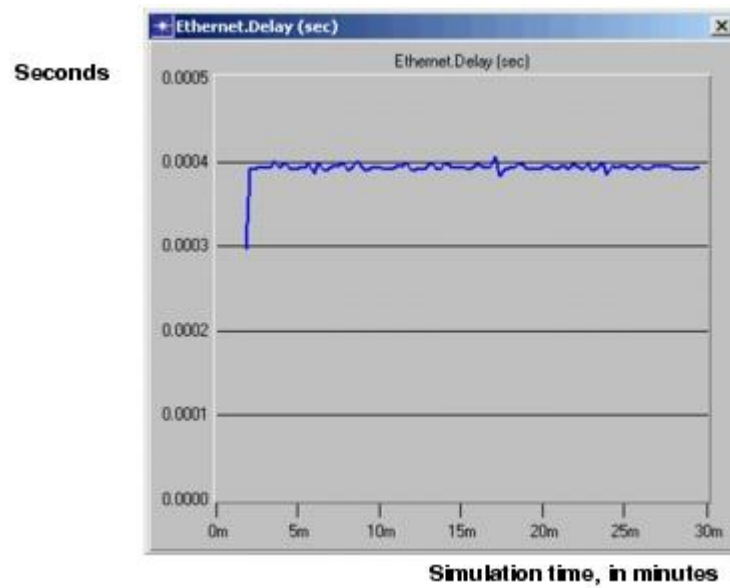
همینطور که می بینید در این لحظه بار سرور ۷,۰۰۰ bits/second است . شما جهت مقایسه گراف در آینده به این گراف احتیاج خواهید داشت .

برای مشاهده Global Ethernet Delay بروی شبکه مطابق روال زیر اقدام فرمائید

۱. بروی محیط کاری Right click کنید
۲. به مسیر Global Statistics > Ethernet > Delay (sec). بروید و چک باکس آن را انتخاب نمائید
۳. به مسیر Object Statistics > Office Network.node_۳۱ > Ethernet > Load (bits/sec) رفته و چک باکس این گزینه را انتخاب نمائید و سپس دکمه Show را جهت مشاهده تاخیر شبکه برای تمام شبکه فشار دهید
۴. مطابق با شکل زیر گزارش را مشاهده خواهید نمود



گراف تاخیر شبکه در Project Editor مطابق شکل زیر نمایش داده خواهد شد .



گسترش شبکه

شبکه فعلی ایجاد و پیاده سازی شد . حالا می بایست شبکه فعلی گسترش بیابد

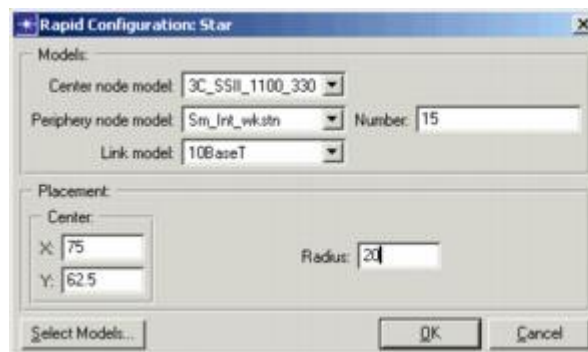
برای دو برابر کردن شبکه مطابق روال زیر اقدام نمائید

۱. منوی **Scenarios > Duplicate Scenario** را انتخاب نمائید
۲. یک نام را جهت نام سناریو جدید انتخاب نمائید
۳. دکمه **Ok** را بزنید
۴. سناریو با تمام نودها ، لینکها ، اطلاعات و آمار و تنظیمات شبیه سازی دوباره ایجاد و با نام جدید ایجاد خواهد شد .
۵. بخش طبقه دوم طبقه اول را دوباره سازی میکند . اما سروری در آن وجود نخواهد داشت .

برای ایجاد بخش جدید مطابق رویه زیر اقدام کنید

۱. منوی **Topology > Rapid Configuration**

۲. مطابق با تعریف در شبکه قبلی اقدام نمائید



۳. دکمه **OK** را بزنید

برای اتصال دو شبکه

۱. در صورتی که پالت اشیا باز نیست دکمه زیر را بزنید



۲. پوشه Cisco ۲۵۱۴ را باز کنید



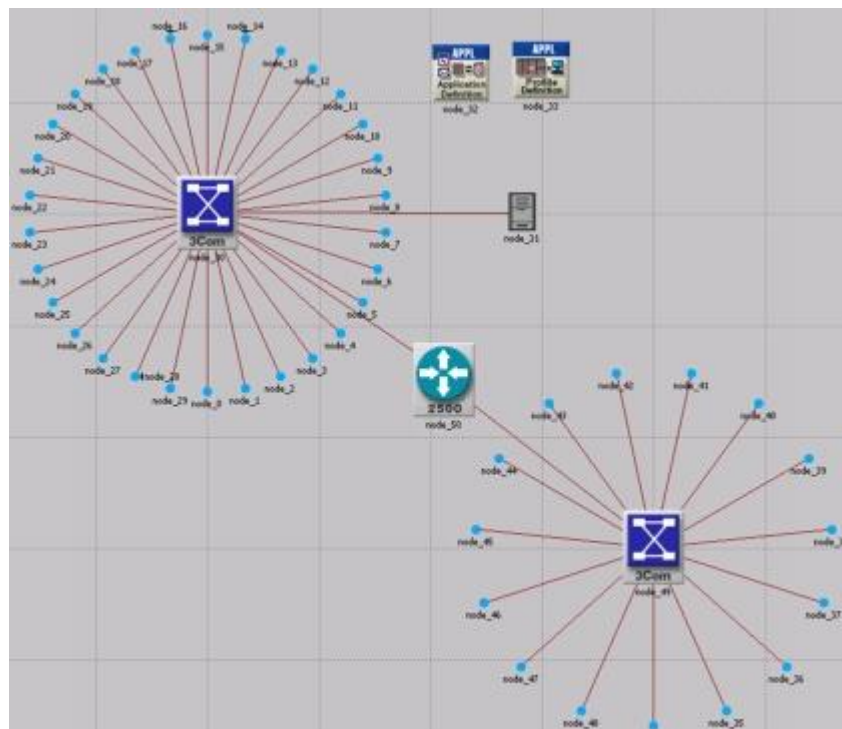
۳. شیء Cisco ۲۵۱۴ را بوی صفحه کار بکشید و بین دو شبکه قرار دهید. دکمه Right click را بزنید

۴. ماژول مدل لینک را باز کرده و بروی BaseT ۱۰ دابل کلیک کنید

۵. روتر را بین دو سوئیچ متصل کرده و Right Click کنید

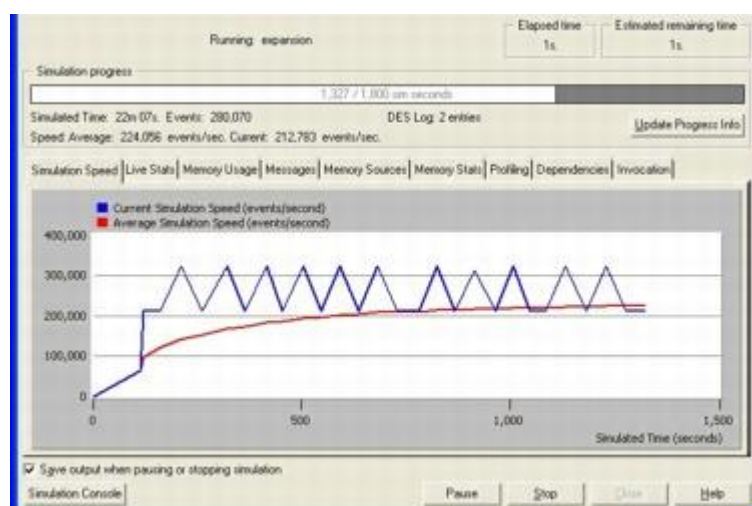
۶. پروژه را Save کنید

۷. پروژه نهایی شبیه شکل زیر خواهد شد



برای اجرای سناریو

۱. منوی DES > Configure/Run Discrete Event Simulation... را انتخاب کنید
۲. در صورت فعال بودن دکمه Detail را فشرده و مقدار Duration را ۰,۵ ساعت بگذارید
۳. مقدار Update interval را ۱۰۰۰۰ بگذارید
۴. دکمه Run را بفشارید



کادر محاوره ای فرایند پیشرفت شبیه سازی

در ابتدا ، مانند قبل یک پنجره پیغام آغاز شبیه سازی را نمایش می دهد و سپس یک گراف متحرک هردو گراف میانگین و نمودار فعلی را در مقیاس رویداد در ثانیه نمایش می دهد و زمانیکه شبیه سازی پایان یافت گراف نتیجه (رویداد بر ثانیه) نمایان خواهد شد

زمانیکه شبیه سازی پایان یافت ، کادر محاوره ای شبیه سازی را ببندید

مقایسه نتیجه:

در پاسخ به سوالهای مطرح شده در مورد دومین شبکه اضافه شده به LAN موجود ، شما نیاز به مقایسه ی نتیجه از دو شبیه سازی هستید .

شما می بایست گزینه **View Results** از آیتم منوی **Object** و **Workspace pop-up** انتخاب کنید تا تفاوت در ترکیب آمار را از سناریوهای شبه به هم گراف بدانید .

برای دیدن سریع لود شدن سرور از هر دو سناریو:

۱. بر روی سرور **node_۳۱** راست کلیک کرده و از منوی **pop-up** گزینه ی **View Results** را انتخاب کنید.

۲. برای نتیجه گزینه ی **Current Project** را از منوی پایین انتخاب کنید.

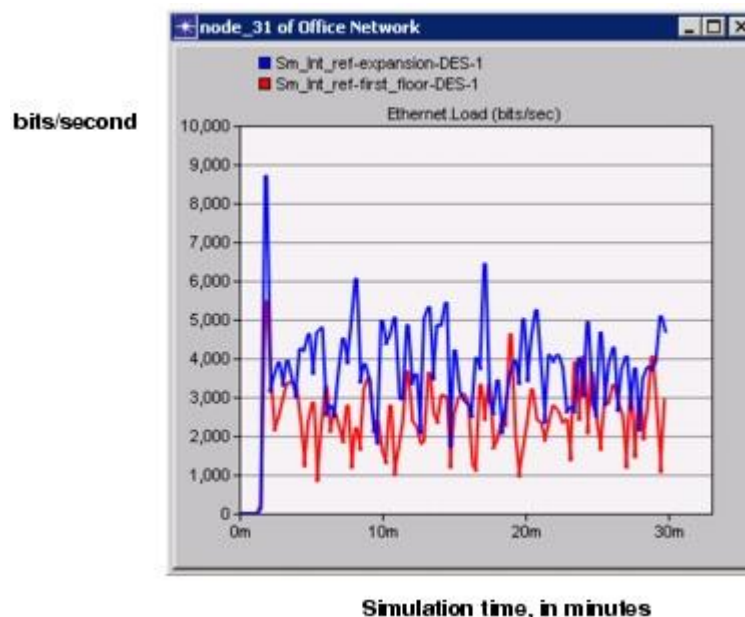
۳. هر دو کادر کنار لیست سناریو را کنترل کنید.

۴. گزینه **Overlaid Statistics** را از منوی کشویی برای نمایش انتخاب کنید

زمانیکه مقایسه نتایج انجام میشود ، انتخاب یک آمار در یک سناریو گراف تولید شده ، مقدار اطلاعات تمام سناریو را نمایش می دهد . برای مشاهده نتیجه

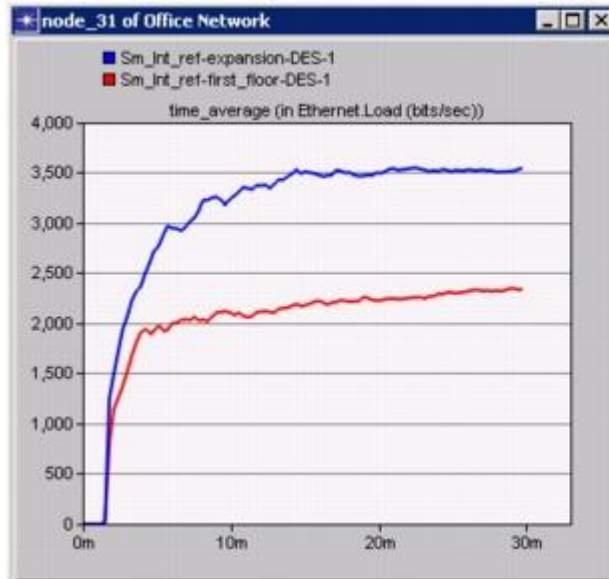
۱. مقدار **Load (bits/sec) > Ethernet > node_۳۱ Office Network** را انتخاب و دکمه **Show** را بزنید . نتیجه درخواستی شما شکل زیر را نمایش خواهد داد .

مقایسه بار شبکه



گراف بعد نمایش دهنده میانگین زمان بار ترافیکی بین دو طبقه است .

مقایسه میانگین زمانی بار ترافیکی سرور



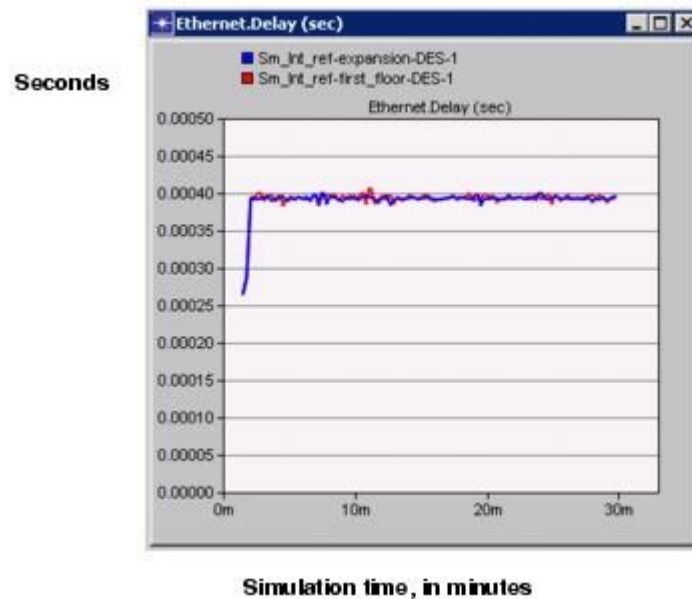
نکته مهم اینکه ، زمانیکه میانگین بار شبکه جدید بیشتر باشد ، تمام بار نمایش داده شده در حالت شبیه سازی شده است و نمایانگر اینست که شبکه پایدار است

Note that while the average load for the expansion scenario is higher (as expected), the load as a whole appears to be leveling off (that is, not monotonically increasing), indicating a stable network.

مرحله آخر نمایش تاخیری که شبکه طبقه دوم به شبکه ما افزوده است . برای مقایسه تاخیر Ethernet باری دو سناری روال زیر را انجام دهید

۱. گراف و پنجره نتیجه گزارش سرور را ببندید
۲. بروی محیط کار click right کرده و گزینه **View Results** را انتخاب کنید
۳. گزینه **Global Statistics -> Ethernet -> Delay (sec)** را انتخاب کنید
۴. دکمه **Show** را بزنید
۵. نتیجه گراف شما شبیه شکل زیر خواهد بود .

مقایسه تاخیر Ethernet



مروری بر پروتکل های MAC در SDH

لایه MAC در پایین بخش لایه پیوند داده است. لایه MAC دسترسی به رسانه ارتباطی را بوسیله تعریف یک مجموعه قوانین که اشتراک موثر و منصف منابع رایج بین چندین کاربر امکان را می دهد را هماهنگ می کند.

بمنظور اجتناب از تصادم در NG-SDH، یک پروتکل کنترل دسترسی به رسانه مطمئن و موثر مورد نیاز است. SDH، vehicle را بعنوان نودهای شبکه دارد و خصوصیات اصلی شان mobility و سرعت بالاست. برنامه های کاربردی Active Safety برای SDH به تنظیم ارتباطات مورداعتماد با حداقل تصادم نیاز دارد. تعداد کمی از پروتکل های MAC برای NG-SDH ها طراحی شده اند که می توانند بصورت موثر برای کار در SDH تطبیق داده شود.

در NG-SDH، نودها خودشان را پیکر بندی کرده و بدون بکارگیری زیر بنای ثابتی یا مدیریت متمرکزی محاوره می کنند. نودهای NG-SDH، فرکانسهای رادیویی که "کانالهای انتقال" نامیده می شوند را به کار می برند که در آن هر کدام به عنوان یک رسانه روی دو ترمینال مجاور که نمی توانند بصورت همزمان انتقال

دهند چون تصادم رخ می دهد. بمنظور اشتراک موثر رسانه ، پرتکلهای MAC به وسیله جامعه تحقیقاتی پیشنهاد شده اند. اشتراک کارا رسانه حتی در SDH بخاطر mobility بالای نودها و تغییرات سریع توپولوژی سخت تر شده است.

SDH شبکه Multi hop است، پرتکلهای MAC در SDH می باید دسترسی به کانال را بین این نودها به منظور بهره وری دسترسی به رسانه به قدر کافی ، هماهنگ کند. بخاطر به اشتراک گذاشتن دسترسی به رسانه انتقالهای همزمان بوسیله دو نود با بیشتر در یک رنج خاص سبب می شود که سیگنالهای تصادف و یا با یکدیگر روبرو می شوند. در حالتی دیگر وضعیت terrain و حرکت نودهای سیار بر طول سیگنال در دسترس دریافت کننده اثر می گذارد. بیشتر نودهای پنهان به انتقال دوباره و رزرو پاکتها منجر خواهد شد. بنابراین آنها با دیگر انتقالهای در حال جریان روبرو خواهند شد و تصادم روی می دهد.

انتخاب یک رنج انتقال رادیویی بهینه یک موازنه بین ماکزیمم بهره وری کانال انتقال و multi hop forwarding است. بنابراین به اجتماع کارایی در ADHOC تأثیر می گذارد.

به هر حال رنج انتقال مؤثر به تعداد نودها در یک منطقه مشخص و موقعیت آنها و سرعت حرکت بستگی دارد. کانالهای بی سیم زمان متغیر و وابسته به موقعیت هستند. نرخ انتقال داده باید با SDH منطبق شوند.

پرتکلهای MAC تجاری مثل TDMA، FDMA، CDMA برای سناریوهای Vehicular مناسب نیستند. شماتیک RTS/CTS، مثل MACA، MACAW، FAMA از لحاظ تئوری برای Broadcast نامناسب اند. در لایه MAC، بهترین چیز سازماندهی دسترسی به رسانه یا کنترل ازدحام است. در طریقه معمول اسلات زمانی و طیف می تواند رزرو شود. برای مثال R-WDM یک پرتوکل Slot Reservation، MAC مشهور است.

پروتکل توکن رینگ بی سیم یک پروتکل کنترل دسترسی به رسانه برای شبکه های بی سیم است (WTRP). WTRP کیفیت سرویس را به معنی latency محدود شده و پهنای باند رزرو شده که در بسیاری از برنامه های کاربردی بلادرنگ بحرانی هستند را ضمانت می کند. احتمال انتقال مناسب در لایه MAC برای همه پیامها برای کاهش احتمال تصادم بسته شناسایی شده است.

برخی کاندیداهای اصلی این پروتکل مثل رویکرد TDMA که ADHOC MAC نامیده می شود در آن رسانه به فریمهای تکراری تقسیم شده و هر فریم خودش به اسلایدهای زمانی مختلف تقسیم شده است. هر نود باید یک اسلاید را برای انتقال که کانال اساسی (BCH) نامیده می شود، برسد.

همچنین یک کنترل دسترسی به رسانه مبتنی بر Directional Antenna است. این یک پروتکل CSMA/CA مبتنی بر D-MAC است.

در SDH، یک کانال انتقال یک رسانه اشتراک گذاشته شده است و بدین منظور، برای اجتناب از تصادم های انتقال وقتی یک نود در حال انتقال روی یک کانال به نودهای مجاورش است قبل از انتقال باید منتظر باشد تا کانال آزاد شود. و چون SDH ساختار ثابتی ندارد برای نودهایش دانستن اینکه رسانه آزاد است یا نه ساده نیست. کارهای زیادی انجام شده و بسیاری برای غلبه بر مسائل Mac های موجود مختلف در SDH جریان است.

WDM، اولین پروتکل پیشنهاد شده برای شبکه های رادیویی بسته ای بود. پروسه WDM مبتنی بر دسترسی تصادفی است وقتی یک نود می خواهد یک کانال را بکار برد اول روی آن انتقال می دهد و سپس اگر تصادم رخ داد برای یک زمان تصادفی قبل از انتقال دوباره منتظر می ماند. حداکثر کارایی این پروتکل ۱۸/۴٪ ظرفیت کانال برای طول پیام ثابت است. دسترسی تصادفی WDM سبب کاهش کارایی می شود. بنابراین ورژن Slotted که S-WDM نامیده می شود، پیشنهاد شد. این پروتکل رسانه را به چندین اسلات زمانی تقسیم می کند و ارسال کننده تلاش می کند که در شروع اسلات زمانی انتقال را انجام دهد

WDM در مقایسه با S-WDM، کارایی سیستم را دو برابر خواهد کرد. دیگر پروتکلی که معرفی شده بود Carrier Sense بود. در CSMA یک نود که قصد ارسال داده دارد اول به کانال گوش می دهد اگر خالی بود ارسال می کند در غیر این صورت یک زمان تصادفی تلاش می کند. Collision Detection، که بعداً برای تشخیص تصادم در طول ارسال اضافه شد، ارسال را متوقف می کند و بعداً تلاش می کند. CSMA/CD هنوز در شبکه های شارژ شده وقتی تعداد زیادی تصادم می تواند انجام گیرد، بهینه نیست ضعف اصلی CSMA/CD این است که مسأله ترمینال های مخفی را حل نمی کند.

دسترسی چندگانه با اجتناب از تصادم (MACA) به مسئله ترمینال مخفی با توافق روی انتقال با مقصد غلبه کرده است. فرستنده با ارسال همگانی بسته RTS، Handshake را مقداردهی می کند. بنابراین همه نودهای

مجاور از شروع انتقال آگاه می باشد. بعد از دریافت بسته RTS، مقصد اگر آماده باشد با ارسال همگانی یک بسته CTS پاسخ می دهد که به همه مجاورها درباره شروع انتقال اطلاع می دهد. با دریافت بسته CTS، فرستنده می تواند شروع به ارسال Unicast بدون هیچ ریسک تصادم بپردازد.

اگر یک نود RTS را دریافت کند اما ان CTS را دریافت نکند می تواند انتقال دهد. به علت مسئله ترمینال بی حفاظ برای امکان به اینکه ترمینالهای بی حفاظ درباره طول زمان انتقال آگاه شوند MACAW افزودن بسته های ACK و DS(Data Sending) را پیشنهاد داد. شکل ۱ تبادل بسته در MACAW شامل بسته های داده و کنترلی را خلاصه می کند. پروتکل های در CDMA چندین کد orthogonal در دسترس هستند. هر نود یک کد را رمزگذاری پیام ها قبل از انتقال آنها بکار می برد.

WDM

WDM یکی از اساسی ترین روش های دسترسی تصادفی به کانال مشترک در شبکه های بی سیم (Mobile Data Network) می باشد. این روش بر اساس ارسال بسته ها توسط پایانه های بی سیم بدون نیاز به هماهنگی با دیگر پایانه ها می باشد. مزیت استفاده از این پروتکل سادگی آن و سهولت بکارگیری آن است بطوریکه هیچ گونه همزمانی بین فرستنده های بی سیم مورد نیاز نمی باشد. پایانه ها بسته های خود را به محض دریافت از لایه بالاتر، ارسال می کنند و در صورت تصادم بسته ها، آنها را مجددا ارسال می نمایند. تصادم بسته ها در طرف گیرنده با روش ساده کد بررسی خطا اشکار می شود. بارزترین عیب این روش، گذردهی پایین به هنگام ترافیک سنگین شبکه می باشد زیرا تعداد تصادم ها با افزایش کاربران شروع به افزایش می کند

پیاده سازی شبیه سازی

در اینجا ما به بررسی پروتکل معروف CSMA می پردازیم که در پیاده سازی این پروتکل از مدل شبکه مشابه استفاده می کنیم. هر مدل شبکه ما، یک مدل نود فرستنده که بسته ها را ارسال می کند و یک مدل نود دریافت کننده را شامل می شوند. مدل پروسس در فرستنده خاص خواهد بود در صورتی که در مدل پروسس دریافت کننده عمومی خواهد بود و تغییر نخواهد کرد.

در طراحی این مدل ها، نودها با هم از طریق یک شبکه غیر متصل به هم متصل می شوند و با استفاده از ماژول پروسس رفتار هر گره را تعریف میکنیم برای مثال فرستنده باید هنگام تصادم بسته آنرا مجددا ارسال

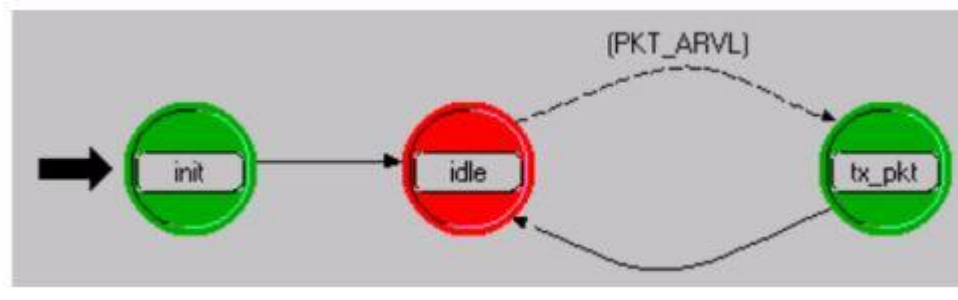
کند. مدل شبکه مورد نظر در برگیرنده مدلی از گره های فرستنده و مدلی از گره گیرنده می باشد و هر گره دارای مدل پروسی است که رفتار گره را تبیین می کند.

پیش از طراحی شبکه بهتر است نگاهی به مدل گره ها و پروس های آن ها به صورت مجزا داشته باشیم. گره فرستنده در WDM اساسا بسته هایی را تولید کرده و با شمارش ، آنها به گذرگاه روانه می کند این عملکرد ساده را می توان در ماژول گره با استفاده از یک منبع ساده و یک پردازشگر پیاده سازی نمود بنابراین با استفاده از ماژول گره یک منبع دیتا و یک گره فرستنده ایجاد می کنیم حال تنها نیاز داریم که پردازشگر، بسته ها را به فرستنده تحویل داده و آنها را شمارش کند نحوه عمل پردازشگر را به هنگام طراحی مدل پروس خواهیم دید.

گره گیرنده اساسا تصادم بسته ها را بر روی گذرگاه چک و بررسی می کند و شامل یک گیرنده گذرگاه و یک پردازشگر می باشد رفتار پردازشگر توسط مدل پروس گیرنده تبیین می شود و مسئولیت آن در این آزمایش اداره کردن بسته های دریافتی به منظور جمع آوری اطلاعات آماری از طریق ایجاد و تغییر بر روی متغیرهای عمومی است . اصولا می توان گفت که این گره تعداد بسته های سالم دریافتی را شمارش کرده و خود بسته ها را دور می ریزد زیرا در حال حاضر محتویات بسته ها اهمیت ندارند.

• طراحی مدل پروس فرستنده

در اینجا مدل پروس فرستنده را ایجاد میکنیم.



مدل پروس فرستنده

۱. گزینه New>Process را از منوی File انتخاب نمائید.

۲. با استفاده از دکمه create state سه حالت برای فرستنده ایجاد کرده ، آنها را به ترتیب به نامهای init و idle (یعنی فرستنده منتظر یک وقفه برای ارسال بسته می باشد) و tx_pkt (یعنی فرستنده در حال ارسال بسته ها می باشد) نامگذاری می کنیم.
۳. فرستنده به هنگام دریافت یک بسته جدید از لایه بالاتر انرا ارسال کرده و سپس به حالت Idle برگشته و به انتظار می نشیند بنابر این حالت Tx_pkt از نوع Forced و حالت Idle از نوع Unforced می باشد. برای این منظور از منوی راست کلیک بر روی حالت Transmit گزینه Make State Forced را انتخاب می کنیم.
۴. انتقال از یک حالت به حالت دیگر را بوسیله دکمه رسم می کنیم. بخاطر داشته باشید که گذر از حالت Idle به Tx_pkt زمانی روی می دهد که یک بسته جدید از لایه بالاتر وارد گردد. از این رو یک انتقال شرطی است و بر روی خط گذر راست کلیک کرده و برای پارامتر توصیفی شرط عبارت Pkt-Arvl تایپ میکنیم. همچنین یک خط انتقال اما بدون شرط از حالت tx_pkt به حالت Idle رسم می نماییم.
۵. مدل پروسس دارای هدری است که در کادر هدر میتوان با برنامه نویسی شرایط گذر را و همچنین متغیرهای غیرمحلی (Global) را تعریف نموده و نحوه جریان بسته ها از منبع دیتا به جریان خروجی از فرستنده را نمایش داد. کادر هدر را باز نموده و کد دستورات زیر را در آن وارد میکنیم:

در Block Header، کد زیر را وارد می کنیم

```
/* Input stream from generator module */

#define IN_STRM •

/* Output stream to bus transmitter module */

#define OUT_STRM •

/* Conditional macros */

#define PKT_ARVL (op_intrpt_type() == OPC_INTRPT_STRM)

/* Global Variable */
```

```
extern int subm_pkts;
```

۶. از آنجایی که می خواهیم شبیه ساز در نقطه ای متوقف گردد برای حداکثر تعداد بسته های ارسالی یک متغیر صحیح max_packet_count در تب متغیر حالت تعریف می کنیم. مقدار آن را هنگامی که که پارامترهای شبیه سازی را مشخص می نماییم تعیین خواهیم کرد. در State Variable Block اطلاعات زیر را وارد می کنیم:

Type	Name	Comments
int	max_packet_count	Number of packets to process

۷. عملیات را بصورت کد، برای حالت init در Executive Block وارد می کنیم (با دوبار کلیک بر روی State):

```
/* Get the maximum packet count, */
```

```
/* set at simulation run-time */
```

```
op_ima_sim_attr_get_int32 ("max packet count",&max_packet_count);
```

۸. تغییرات را ذخیره می کنیم.

۹. عملیات را به صورت کد، برای حالت Tx_pkt در Executive Block وارد می کنیم:

```
/* Outgoing packet */
```

```
Packet *out_pkt;
```

```
/* A packet has arrived for transmission. Acquire */
```

```
/* the packet from the input stream, send the packet */
```

```
/* and update the global submitted packet counter. */
```

```
out_pkt = op_pk_get (IN_STRM);
```

```
op_pk_send (out_pkt, OUT_STRM);
```

```
++subm_pkts;
```

```
/* Compare the total number of packets submitted with */
```

```
/* the maximum set for this simulation run. If equal */
```

```
/* end the simulation run. */
```

```
if (subm_pkts == max_packet_count)
```

```
{
```

```
op_sim_end ("max packet count reached.", "", "", "");
```

```
}
```

۱۰. تغییرات را ذخیره می کنیم.

حالت Tx_pkt بسته ها را از جریان ورودی دریافت کرده و آنها را ارسال می کند و همچنین متغیر شماره بسته های ارسالی را بروز کرده و در صورت آن که این شمارنده به مقدار max_packet_count برسد عملیات شبیه سازی را به پایان می رساند بنابراین در کادر اجرایی ورود به این حالت دستورات زیر را وارد میکنیم : متغیر subm_pkts یک متغیر global است که در پروسس گیرنده تعریف خواهد شد.

۱۱. در این مرحله برای آن که شبیه ساز بداند چه زمانی کار خود را به اتمام برساند لازم است مقداری را به متغیر max_packet_count نسبت دهیم برای این منظور باید یک پارامتر توصیفی (Attribute) تعریف کرده و آن را به هنگام راه اندازی شبیه ساز مقدار دهی نماییم به گونه ای که مقدار این پارامتر به متغیر حالت max_packet_count بارگذاری گردد. برای این منظور از منوی Interface ماژول پروسس گزینه Process Interfaces را انتخاب کرده و در آن عبارت "max packet count" را از نوع عدد صحیح تعریف می نماییم.

Attribute Name	Group	Type	Units	Default Value
max packet count		integer		0

۱۲. در پایان لازم است که رابط (Interface) ماژول پروسس را کمی ویرایش نماییم به این ترتیب که گزینه ProcessInterface را از منوی Interface انتخاب کرده و پارامتر توصیفی

begsim_intrpt را فعال (enable) کرده و وضعیت تمام پارامترها را به شکل hidden تعریف می نماییم.

پس از مراحل بالا باید مدل پروسس را کامپایل کنیم برای این منظور بر روی دگمه کلیک نموده و پس از آن مدل پروسس را تحت عنوان WDM_tx ذخیره می کنیم.

• طراحی مدل گره فرستنده

۱. از منوی File گزینه New>Node Model را انتخاب کرده و دکمه Ok را کلیک کنید.

۲. دو مدل پردازشگر (processor) و یک فرستنده ایجاد کرده و نام آنها را به ترتیب "gen" و "tx_proc" و نام فرستنده را "radio_tx" می گذاریم.

۳. مدل gen را به tx_proc و مدل tx_proc را به فرستنده به وسیله جریان بسته ها packet streams به یکدیگر متصل می کنیم.

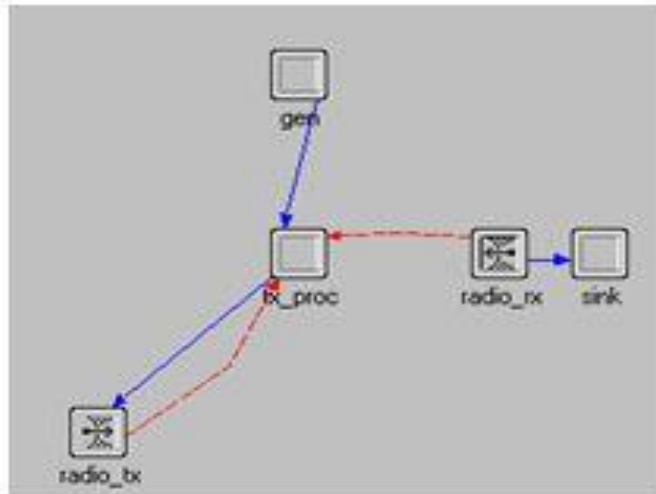
۴. پارامتر توصیفی مدل پروسس gen را به صورت پیش فرض simple_source و از آن مدل tx_proc را به مدل پروسسی که در قسمت بالا طراحی گردید یعنی WDM_tx نسبت می دهیم.

۵. در مدل فرستنده مطلوب است که فاصله زمانی بین بسته های وارد شده "packet Interarrival Time" از سوی gen را با مقادیر مختلفی تنظیم و آزمایش نمائیم برای این منظور می توانیم مقدار دهی این پارامتر را در سطح بالاتری انجام دهیم از این روی بر روی این پارامتر در مدل gen راست کلیک کرده و گزینه "promote to higher layer" را انتخاب میکنیم.

۶. با این که تا این مرحله برای WDM کافی است از آن جا که این مدل نود را برای مدل پروسس CSMA نیز به کار می بریم آن را را کامل تر طراحی می کنیم:

۷. بدین منظور دو ماژول processor جدید با نام های sink و radio_rx اضافه می نماییم.

آن ها را بصورت زیر به هم متصل می کنیم:

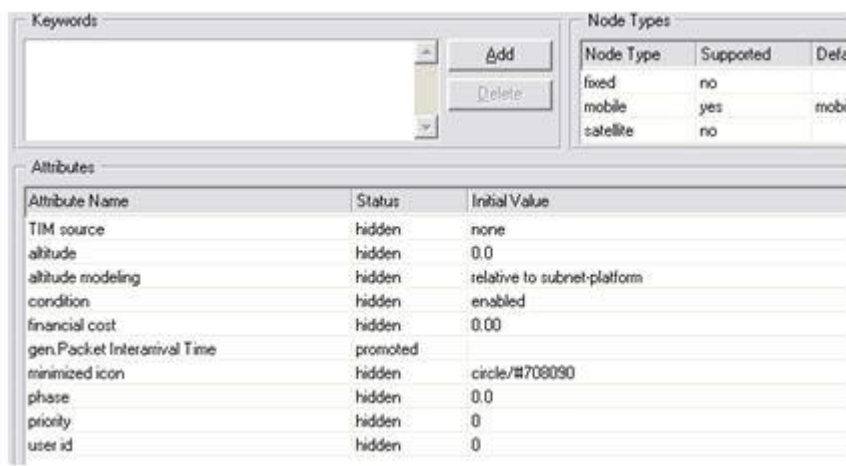


۸. با بکارگیری Create Statistic wire بین tx_proc و Radio_tx ، اتصال برقرار می کنیم. در attribute آن، rising Edge Trigger را disable و Falling Edge Trigger را disable می نماییم همچنین اتصالات را به منظور اینکه به درستی متصل شده باشند، بررسی می کنیم و در صورت مشکل آن ها را اصلاح می کنیم:

ارتباط Tx_proc و gen از Src Stream به [0] Stream src

ارتباط radio_rx ، tx_proc از dest stat به [0] Instat

۹. Node Interfaces > Node Type را انتخاب کنید و Mobile را انتخاب کنید. همه Status ها را به hidden تغییر دهید به جز gen..packet Interarrival Time که Permoted باشد.



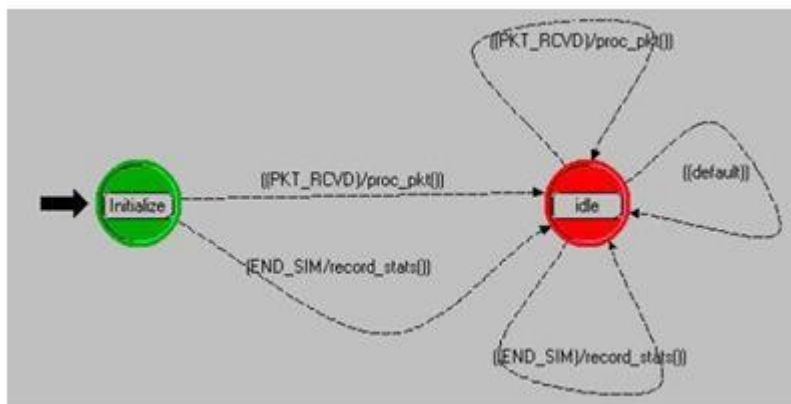
۱۰. در پایان مدل گره فرستنده تحت عنوان `cct_tx` ذخیره می کنیم.

• طراحی مدل پروسس گیرنده

۱. از منوی `File` گزینه `New>Process Model` را انتخاب و بر روی دکمه `Ok` کلیک می کنیم.

۲. دو حالت `state` ایجاد کرده و آن ها را به `"init"` و `"idle"` نامگذاری می کنیم. حالت `init` را با انتخاب گزینه `make state forced` از منوی راست کلیک تغییر می دهیم.

۳. برای رسم خطوط گذر از یک حالت به حالت دیگر باید عملکرد هر حالت و شرایط گذر را درک نماییم. اساسی ترین عمل گیرنده این است که در انتظار رسیدن بسته باشد و با رسیدن بسته مشخصه آماری مربوطه را بروز نماید. این رفتار به خوبی در شکل مدل پروسس ذیل دیده میشود. هنگامی که شبیه سازی اجرا می شود گیرنده به حالت `init` رفته و یک مقدار اولیه به تمام پارامترهای شبیه سازی می دهد (مانند `max_packet_cout=۰`). با ورود یک بسته (شرط `pkt_rcvd`) پروسس، تابع `proc_pkt()` را اجرا کرده و به حالت `idle` منتقل میشود. تابع `proc_pkt()` در واقع بسته را دریافت کرده و شمارنده بسته را یک واحد افزایش داده و سپس بسته را دور میریزد. گیرنده زمانی که به حالت `idle` وارد شد در آنجا منتظر ورود بسته و یا فرمان خاتمه شبیه سازی می باشد. گذر از حالت `idle` به `init` تنها زمانی اتفاق می افتد که شبیه سازی به پایان برسد (یعنی شرط `END_SIM`). در پایان شبیه سازی، تمام آمارها بوسیله روال `record_stats()` جمع آوری می شود.



۴. مطابق شکل فوق شرایط انتقال را با تغییر پارامتر **condition** خطوط گذر و توابعی که باید اجرا گردند با تغییر پارامتر **executive** به مقادیری که در شکل دیده می شود ویرایش می کنیم.

۵. در کادر هدر مدل پروسس دستورات زیر را وارد می کنیم.

```
/* Input stream from radio receiver */

#define IN_STRM ,

/*Conditional macros */

#define PKT_RCVD (op_intrpt_type () == OPC_INTRPT_STRM)

#define END_SIM (op_intrpt_type () == OPC_INTRPT_ENDSIM)

/* Global variable */

int subm_pkts = ۰;
```

در مدل پروسس گیرنده با استفاده از متغیر حالت **rcvd_pkts** تعداد بسته های معتبر را شمارش می کنیم از این روی کادر **statevariables** را باز کرده و متغیر **rcvd_pkts** را از نوع **integer** تعریف می کنیم.

Type	Name	Comments
int	rcvd_pkts	Received packet counter

۶. در مدل گیرنده از تابع `proc_pkt()` برای دریافت بسته و از تابع `record_stats()` برای نوشتن آمار بسته ها استفاده گردیده است بنابر این کد توابع اخیر را باید در کادر `functionblock` مدل بنویسیم. کادر مذکور را باز کرده و دستورات زیر را وارد می کنیم:

```
/* This function gets the received packet, destroys */
```

```
/* it, and logs the incremented received packet total*/
```

```
static void proc_pkt (void)
```

```
{
```

```
Packet* in_pkt;
```

```
FIN (proc_pkt());
```

```
/* Get packet from bus receiver input stream */
```

```
in_pkt = op_pk_get (IN_STRM);
```

```
/*Destroy the received packet */
```

```
op_pk_destroy (in_pkt);
```

```
/* Increment the count of received packet */
```

```
++rcvd_pkts;
```

```
FOUT;
```

```
}
```

```
/* This function writes the end-of-simulation channel */
```

```
/* traffic and channel throughput statistics to a */
```

```
/* scalar file */
```

```
static void record_stats (void)
```

```
{
```

```

double cur_time;

FIN (record_stats());

cur_time = op_sim_time();

/* Record final statistics */

op_stat_scalar_write ("Channel Traffic G",
(double) subm_pkts / cur_time);
Bding the WDM
op_stat_scalar_write ("Channel Throughput S",
(double) rcvd_pkts / cur_time);

FOUT;

}

```

همانطور که مشاهده می شود ترافیک کانال **channel load** برابر با تعداد بسته های ارسالی در مدت شبیه سازی و گذردهی کانال **channel throughput** برابر با تعداد بسته های سالم دریافت شده در همین فاصله زمانی می باشد.

پس از تعریف متغیرها و توابع مورد نیاز مدل پروسس ، تنها کد اجرایی ورود به حالت **init** می ماند. بر روی نیمه بالایی حالت **init** دوبار کلیک کرده و سپس کد زیر را وارد میکنیم:

```

/* Initialize accumulator */

rcvd_pkts = 0;

```

۷. مانند مدل پروسس فرستنده در اینجا نیز باید رابط (**interface**) مدل گیرنده را ویرایش نمائیم از منوی **Interface** گزینه **Process Interface** را انتخاب کرده و پارامتر **endsim intrpt** را فعال (**enable**) نموده و پارامتر **status** را به **hidden** تغییر می دهیم.
۸. در پایان مدل را کامپایل کرده و با نام **cct_rx** ذخیره می کنیم.

• طراحی مدل گره گیرنده

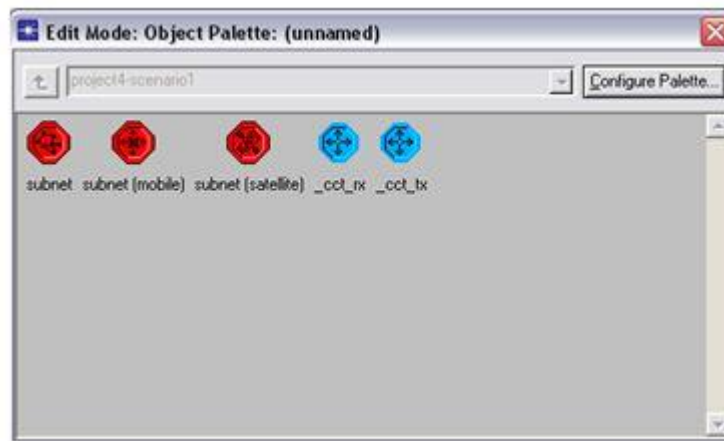
۱. از منوی File گزینه New>Node Model را انتخاب کرده و دکمه Ok را کلیک نمایید.
۲. یک مدل پردازشگر و یک مدل گیرنده ایجاد کرده و نام آنها را به ترتیب rx_proc و radio_rx می گذاریم.
۳. گیرنده radio_rx را به پردازشگر rx_proc بوسیله دکمه (packet stream) جریان بسته متصل می کنیم.
۴. با بکارگیری Create Statistic wire بین tx_proc و Radio_rx، اتصال برقرار می کنیم. در attribute آن، rising Edge Trigger و Falling Edge Trigger، disable می باشند.
۵. کادر تبدیلی Attribute مدل پردازشگر rx_proc را باز کرده و به پارامتر مدل پردازش آن، مدل قسمت قبلی را یعنی cct_rx نسبت می دهیم.
۶. رابط (interface) گره را از منوی Interface تغییر می دهیم به این ترتیب که در جدول NodeType مقدار پارامتر به no برای نوع mobile تغییر داده و yes می کنیم و همچنین پارامترهای Status را به hidden تغییر می دهیم.
۷. مدل گره را تحت عنوان cct_rx ذخیره می کنیم.



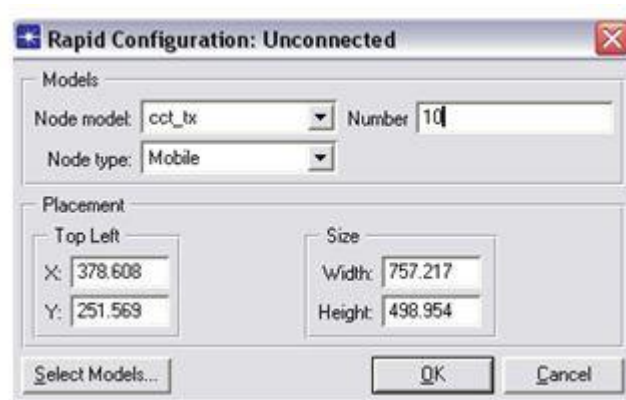
در اینجا طراحی مدل گره و پروسس WDM به اتمام می رسد حال باید مدل شبکه را ایجاد کنیم بدین منظور بدین ترتیب عمل می کنیم:

طراحی مدل شبکه WDM

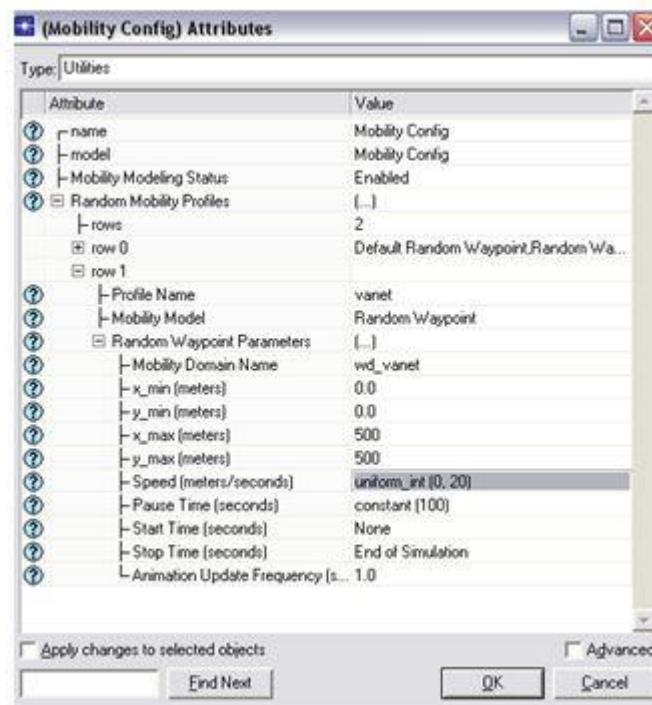
۱. از منوی File گزینه New>Project را انتخاب کرده و دکمه Ok را کلیک می کنیم.
۲. نام پروژه را cct_Network نامیده و سناریو آن را WDM انتخاب کرده و با برنامه wizard مشخصات شبکه را به شکل زیر تعیین می کنیم.
۳. Enterprise > ... > NG-SDH و سپس کلید Ok را کلیک می کنیم.
۴. در پالت object ابتدا Configur Palette>Clear و سپس آیتم های cct_tx و cct_rx را به پالت اضافه می کنیم و آن مدل شبکه را با نام cct ذخیره می کنیم. این ها که به صورت نود موبایل اضافه می شوند امکان ایجاد مدل شبکه را فراهم می کنند.



۵. در پنجره Project از منوی Topology گزینه Rapid configuration>Unconnected را انتخاب کرده و مطابق شکل زیر مقادیر کادر تبادلی را پر می کنیم.



۶. حال برای تعیین سرعت برای نودهای موبایل بدین صورت عمل کرده که از پالت mobility Config را انتخاب کرده و در قسمت Random Mobility File ، نام Profile را تعیین کرده (SDH) و مدل حرکتی را Radom Waypoint قرار میدهم و در Radom Waypoint Parameter پارامتر Speed را با سرعتی بالا مقدار دهی می کنیم (خصوصیات خاص SDH در مقابل NG-SDH درضمن بدلیل اینکه شبکه SDH است دامنه ای برای حرکت نود سیار ایجاد می کنیم (نودهای SDH در محدوده خاص حرکت می کنند) برای این منظور mobilityDomain را انتخاب کرده و نودها را در بازه های خاص محدود کرده و نام این دامنه را در mobility Domain Name انتخاب می کنیم.



۷. بدین وسیله با این توپولوژی ۱۰ فرستنده ایجاد کرده و لازم است که گره گیرنده را به ان اضافه کنیم. بدین منظور یک گره cct_rx به صفحه اضافه می کنیم.
۸. پس از اتمام مراحل بالا ، مدل را با نام شبکه cct_network ذخیره می کنیم .

• اجرای شبیه سازی WDM

در این بخش گذرده‌ی پروتکل WDM را در شرایط مختلف ترافیک کانال مورد تحلیل قرار می‌دهیم. ۱۲ شبیه‌سازی مختلف همزمان بوسیله ادیتور شبیه‌ساز انجام می‌گیرد. پارامتر فاصله زمانی دریافت بسته (inter-arrival time) توسط فرستنده را طی چندین شبیه‌سازی تغییر می‌دهیم تا سطوح مختلفی از ترافیک ایجاد گردد و همچنین با تغییر سرعت‌ها و تعداد نودهای موبایل شبکه را تحلیل می‌کنیم.

۱. Import > Scenario Components > Scenario را انتخاب کنید.
۲. SequenseSimulation را از منو انتخاب کرده و cct_network-CSMA را انتخاب کنید.
۳. از منو DES گزینه (Advanced) Configure Discrete Event Simulation را انتخاب می‌کنیم.
۴. راست کلیک کرده و گزینه Edit Attributes را انتخاب می‌کنیم.
۵. روی نود درخت Execution کلیک کرده و Advanced را انتخاب کنید.
۶. مدل شبکه را cct_network-WDM انتخاب کنید.
۷. روی نود Outputs و سپس Statistic Collection کلیک کنید.
۸. Prob File را به <None> تنظیم کنید. تابع op_stat_Scalar_Write و فایل Scalar جایگزین می‌شود.
۹. فایل Scalar را به cct_a تنظیم کنید.

این فایل خروجی تابع op_stat_Scalar_Write را که Function Block در cct_rx وارد کردید، جمع‌آوری می‌کند.

۱۰. Clear Scalar File Before Running Simulation Set را تیک بزنید.
۱۱. بر روی تب Global Attributes کلیک کرده و مقدار فیلد maxpacketcount را به ۱۰۰۰ تنظیم می‌کنیم.
۱۲. بر روی ObjectAttributes کلیک کرده و دقت کنید ۱۲ مقدار در Network.*.gen.PacketInterarrivalTime موجود باشد.
۱۳. Yes را کلیک کرده و کادر SimulationSet را ببندید.
۱۴. File > Save را انتخاب کنید.

۱۵. بر روی دکمه نوار ابزار **Execute Simulation Sequence** کلیک کرده و عملیات اجرا را تایید می کنیم.

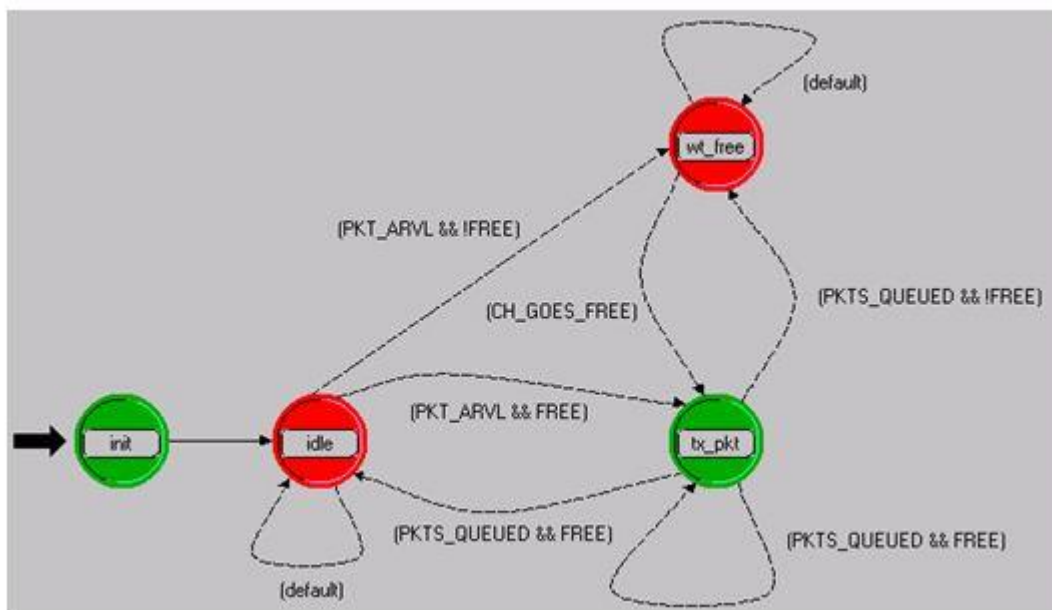
۱۶. وقتی شبیه سازی تکمیل شد کادر **Simulation Sequence** را می بندیم.

CSMA

کارایی پروتکل دسترسی تصادفی **WDM** می تواند وسیله افزودن قابلیت **CarrierSense** پشتیبانی شود. این قابلیت در پروتکل **CSMA** کلاسیک که نیاز به یک منبع داده برای گوش دادن به کانال و مشخص کردن آزاد بودن کانال بمنظور ارسال داده را دارد. بدین منظور می توان از پروسس **WDM_tx** موجود را بعنوان این که پروسس منتظر شود تا کانال آزاد شود بکار برد.

در **PrjectEditor** مدل **WDM_tx** را انتخاب کنید.

حالت ها و گذرها را اصلاح کنید تا شکل زیر حاصل شود.



۱. یک وضعیت جدید ایجاد کرده و آن را **Wt_free** نام دهید.

۲. یک گذر از **wt_free** به **tx_pkt** ایجاد کنید و وضعیت را به **CH_GOES_FREE** تغییر دهید.

۳. یک گذر از حالت **Wt_free** به خودش برگردانید و وضعیت را به **default** تنظیم کنید.

۴. یک گذر از idle به wt_free ایجاد کنید و condition آن را به PKT_ARVL&&FREE تغییر دهید.

۵. یک گذر از idle به خودش برگردانید و وضعیت را به default تنظیم کنید.

۶. وضعیت گذر از idle به tx_pkt را به PKT_ARVL&&FREE تغییر دهید.

۷. وضعیت گذر غیر شرطی از tx_pkt را به شرطی با تنظیم وضعیت به default تغییر دهید.

۸. یک گذر از حالت tx_pkt به خودش برگردانید و وضعیت را به

PKTS_QUEUED&&FREE تنظیم کنید.

۹. در آخر، یک گذر از tx_pkt به wt_free ایجاد و وضعیت را

PKTS_QUEUED&&FREE تنظیم کنید.

ویرایش Header Block

Header Block را می باید به منظور اینکه پروسس قبل از انتقال ، بررسی کند که کانال آزاد است یا نه ، تغییر دهید. برای اینکه پروسس بسته را ارسال کند ، باید اول با پروسیجر هسته op_stat_local_read() با خواندن آمارهای busy کانال چک کند که کانال آزاد است. اگر کانال آزاد نیست. پروسس به وضعیت wt_free وارد می شود تا اینکه وقفه "channel goes free" در یافت شود.

در سطح نود ، statisticwire وقتی busy به ۰,۰ تغییر کند تریگر شده است و بوسیله enable کردن ویژگی Fallingedge trigger فعال شده است .

خطوط زیر را در پایان Header Block مدل پروسس اضافه کنید.

```
/* input statistic indices */
```

```
#define CH_BUSY_STAT ۰
```

```
/* Conditional macros */
```

```
#define FREE (op_stat_local_read (CH_BUSY_STAT) == ۰,۰)
```

```
#define PKTS_QUEUED (!op_strm_empty (IN_STRM))
```

```
#define CH_GOES_FREE (op_intrpt_type () ==  
OPC_INTRPT_STAT)
```

۱. Header Block را ذخیره کنید.

۲. File>Save as را انتخاب کنید و مدل را به csmatx تغییر نام دهید.

۳. مدل را کامپایل کنید و Proccess Editor را ببندید.

برای پشتیبانی مدل نود ارسال کننده برای پشتیبانی CSMA:

۱. File > Recent Files > Node Model را انتخاب کنید و ccttx را انتخاب کنید.

۲. روی statisticwire راست کلیک کرده و Edit را انتخاب کرده و fallingEdgetrigger را Enable کنید و Ok را کلیک کنید.

۳. کادر تبادلی Attribute برای مازول txproc را باز کنید و مدل پروسس را به csmatx تغییر داده کادر را ببندید.

پروسسور اکنون یک مدل پروسس که روی وقفه های آماری شلوغی کانال ارسال شده روی مازول دریافت کننده عمل می کند را بکار می برد.

۴. File>Save as را انتخاب کنید و انرا به مدل cctcsmatx تغییر نام دهید و NodeEditor را ببندید

تعریف مدل شبکه

حال که مدلها برای پشتیبانی CSMA اصلاح شد، نیاز به تغییر مدل شبکه برای استفاده از مدل‌های جدید است. به جای ایجاد یک مدل جدید، می‌توانید سناریوی جدید را duplicate کنید و تغییرات دلخواه را اعمال کنید.

۱. در Project Editor...>Duplicate Scenario... را انتخاب کرده و نام جدید سناریو

را CSMA انتخاب کنید.

تنها تغییر مدل شبکه استفاده از نود انتقال دهنده جدید CSMA است.

۲. مدل نود cct_csma_tx را به پالت اضافه کرده و پالت را با نام پیش فرض ذخیره کنید.

۳. روی نودهای فرستنده راست کلیک کرده و Select Similar Nodes را انتخاب کنید.

۴. روی همه نودهای انتخاب شده کلیک کرده و Edit Attributes را از منو انتخاب کنید.

۵. Apply Change To Selected Objects را تیک بزنید.

۶. خصوصیت model را به cct_csma_tx تغییر داده و Ok را کلیک کنید.

یک کادر تبدلی باز می‌شود و هشدار می‌دهد که تغییرات نمی‌تواند نیمه کاره باشد.

۷. Yes را کلیک کنید.

مدل‌های نود به cct_csma_tx تغییر داده می‌شوند.

پیکربندی شبیه‌سازی CSMA

۱. از منو DES گزینه Configure Discrete Event Simulation(Advanced) را انتخاب می‌کنیم.

۲. روی Simulation Set و Edit Attributes را انتخاب کنید.

۳. seed را به ۱۱ تغییر دهید.

۴. روی نود Outputs و سپس Statistic Collection کلیک کنید.

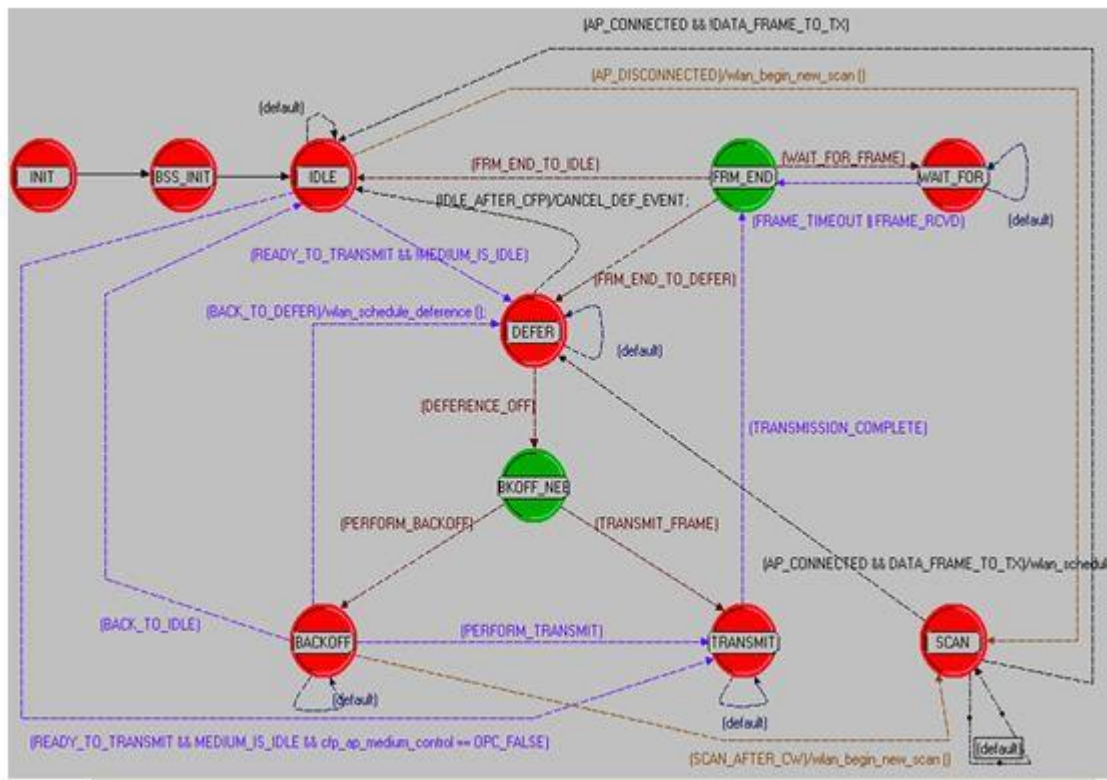
۵. فایل Scalar را به cct_c تنظیم کنید.

۶. Prob File را به <None> تنظیم کنید.

۷. Yes را کلیک کرده و کادر Simulation Set را ببندید.
۸. File > Save را انتخاب کنید.
۹. بر روی دکمه نوار ابزار Execute Simulation Sequence کلیک کرده و عملیات اجرا را تایید می کنیم.
۱۰. وقتی شبیه سازی تکمیل شد کادر Simulation Sequence را می بندیم.

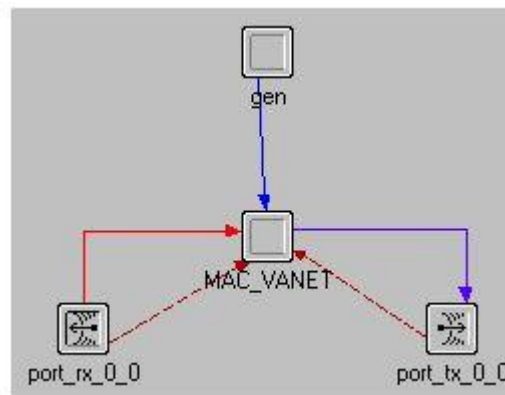
بررسی پروتکل MAC دیگر در شبیه سازی SDH

طراحی مدل پروسس



۱. از منوی File گزینه New>ProcessModel را انتخاب و بر روی دکمه Ok کلیک می کنیم.
۲. ده حالت state ایجاد کرده و آنها را نامگذاری می کنیم. حالت های سبز از نوع Forced و قرمز از نوع Unforced هستند که با انتخاب گزینه make state forced از منوی راست کلیک تغییر می دهیم.
۳. با رسم خطوط گذر از یک حالت به حالت دیگر عملکرد هر حالت و شرایط گذر را ایجاد می نماییم.
۴. مطابق شکل فوق شرایط انتقال را با تغییر پارامتر condition خطوط گذر و توابعی که باید اجرا گردند با تغییر پارامتر executive به مقادیری که در شکل دیده می شود ویرایش می کنیم.
۵. در کادر DB، FB، SV، HB و TV در مدل پروسس دستورات مورد نیاز را وارد می کنیم.
۶. پس از مراحل بالا باید مدل پروسس را کامپایل کنیم برای این منظور بر روی دکمه کلیک نموده و پس از آن مدل پروسس را تحت عنوان MAC ذخیره می کنیم.

طراحی مدل گره



۱. دو مدل پردازشگر (Processor) و دو فرستنده ایجاد کرده و نام آنها را به ترتیب "gen" و "MAC_SDH" و نام فرستنده را "port_tx_۰_۰" و "port_rx_۰_۰" می گذاریم.
۲. مدل gen را به MAC_SDH و این مدل را به فرستنده و گیرنده بوسیله جریان بسته ها packet streams به یکدیگر متصل می کنیم.
۳. پارامتر توصیفی مدل پروسس gen را به صورت پیش فرض simple_source و از آن مدل MAC_SDH را به مدل پروسسی که در قسمت بالا طراحی گردید یعنی MAC نسبت می دهیم.

۴. در مدل فرستنده مطلوب است که فاصله زمانی بین بسته های وارد شده "packet Interarrival Time" از سوی gen را با مقادیر مختلفی تنظیم و آزمایش نمائیم برای این منظور می توانیم مقدار دهی این پارامتر را در سطح بالاتری انجام دهیم از این روی بر روی این پارامتر در مدل gen راست کلیک کرده و گزینه "promote to higher layer" را انتخاب می کنیم.
۵. با بکارگیری Create Statistic wire بین MAC_SDH و "port_rx_0_0" و "port_tx_0_0"، اتصال برقرار می کنیم.
- در attribute فرستنده، rising Edge Trigger را disable و Falling Edge Trigger را enable می نماییم و همچنین در attribute گیرنده، rising Edge Trigger و Falling Edge Trigger را disable می نماییم همچنین اتصالات را به منظور این که درستی متصل شده باشند را بررسی می کنیم و در صورت مشکل آنها را اصلاح می کنیم.
۶. interfaces>NodeInterfaces را انتخاب کنید و NodeType را Mobile انتخاب کنید. همه Status ها را به hidden تغییر دهید به جز gen..packet Interarrival Time که Promoted باشد.
۷. در پایان مدل گره فرستنده تحت عنوان دلخواه مثل SDH ذخیره می کنیم.

طراحی مدل شبکه پروتکل مورد نظر

۱. از منوی File گزینه New>Project را انتخاب کرده و دکمه Ok را کلیک می کنیم.
۲. نام پروژه را SDH_MAC نامیده و سناریو آن را MAC انتخاب کرده و با برنامه wizard مشخصات شبکه را به شکل زیر تعیین می کنیم
۳. NG-SDH > ... > Enterprise و سپس کلید Ok را کلیک می کنیم.
۴. در پالت object ابتدا Configur Palette و سپس آیت SDH را به پالت اضافه می کنیم و آن مدل شبکه را با نام دلخواه ذخیره میکنیم. این ها که به صورت نود موبایل اضافه می شوند امکان ایجاد مدل شبکه را فراهم میکنند.
۵. در پنجره Project از منوی Topology گزینه Rapid configuration>UnconnectedNet را انتخاب کرده در کادر تبدیلی Node model را SDH و تعداد را تعیین و نوع نود را mobile انتخاب می کنیم .

۶. حال برای تعیین سرعت برای نودهای موبایل بدین صورت عمل کرده که از پالت mobility Config را انتخاب کرده و در قسمت Random Mobility File، نام Profile را تعیین کرده (SDH) و مدل حرکتی را Radom Waypoint قرار می دهیم و در Radom WaypointParameter پارامتر Speed را با سرعتی بالا مقدار دهی می کنیم (خصوصیات خاص SDH در مقابل

۷. NG-SDH) درضمن بدلیل اینکه شبکه SDH است دامنه ای برای حرکت نود سیار ایجاد می کنیم (نودهای SDH در محدوده خاص حرکت می کنند) برای این منظور mobility Domain را انتخاب کرده و نودها را در بازه های خاص محدود کرده و نام این دامنه را در mobility DomainName انتخاب می کنیم.

۸. بدین وسیله با این توپولوژی تعدادی نود ایجاد کرده ایم. پس از اتمام مراحل بالا، مدل را ذخیره می کنیم.

۹. از منو DES گزینه (Advanced) Configure Discrete Event Simulation را انتخاب می کنیم.

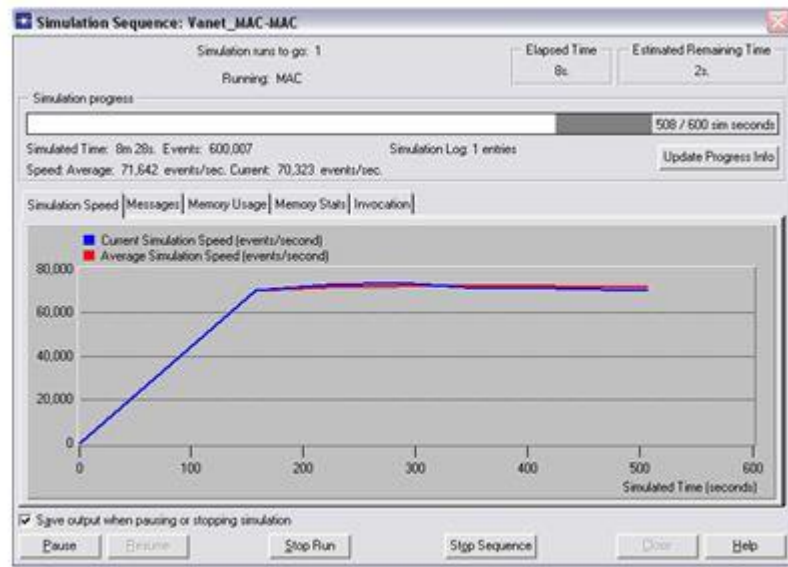
۱۰. راست کلیک کرده و گزینه Edit Attributes را انتخاب می کنیم و تغییرات لازم را اعمال می کنیم.

۱۱. Yes را کلیک کرده و کادر SimulationSet را ببندید.

۱۲. File>Save را انتخاب کنید.

۱۳. بر روی دکمه نوار ابزار ExecuteSimulationSequence کلیک کرده و عملیات اجرا را تأیید می کنیم.

۱۴. وقتی شبیه سازی تکمیل شد کادر SimulationSequence را می بندیم.



جزئیات نصب نرم افزار Opnet

این صفحات شامل آموزش تصویری و قدم به قدم نصب و تنظیم نرم افزار OPNET است

نکته : قبل از نصب نرم افزار OPNET از نصب یک کامپایلر C بروی دستگاه خود مطمئن شوید . هرچند شما میتوانید پس از عملیات نصب نیز کامپایلر را بروی دستگاه خود نصب نمایید .

برای استفاده از نرم افزار OPNET سه مازول زیر می بایست نصب و پیکر بندی گردند

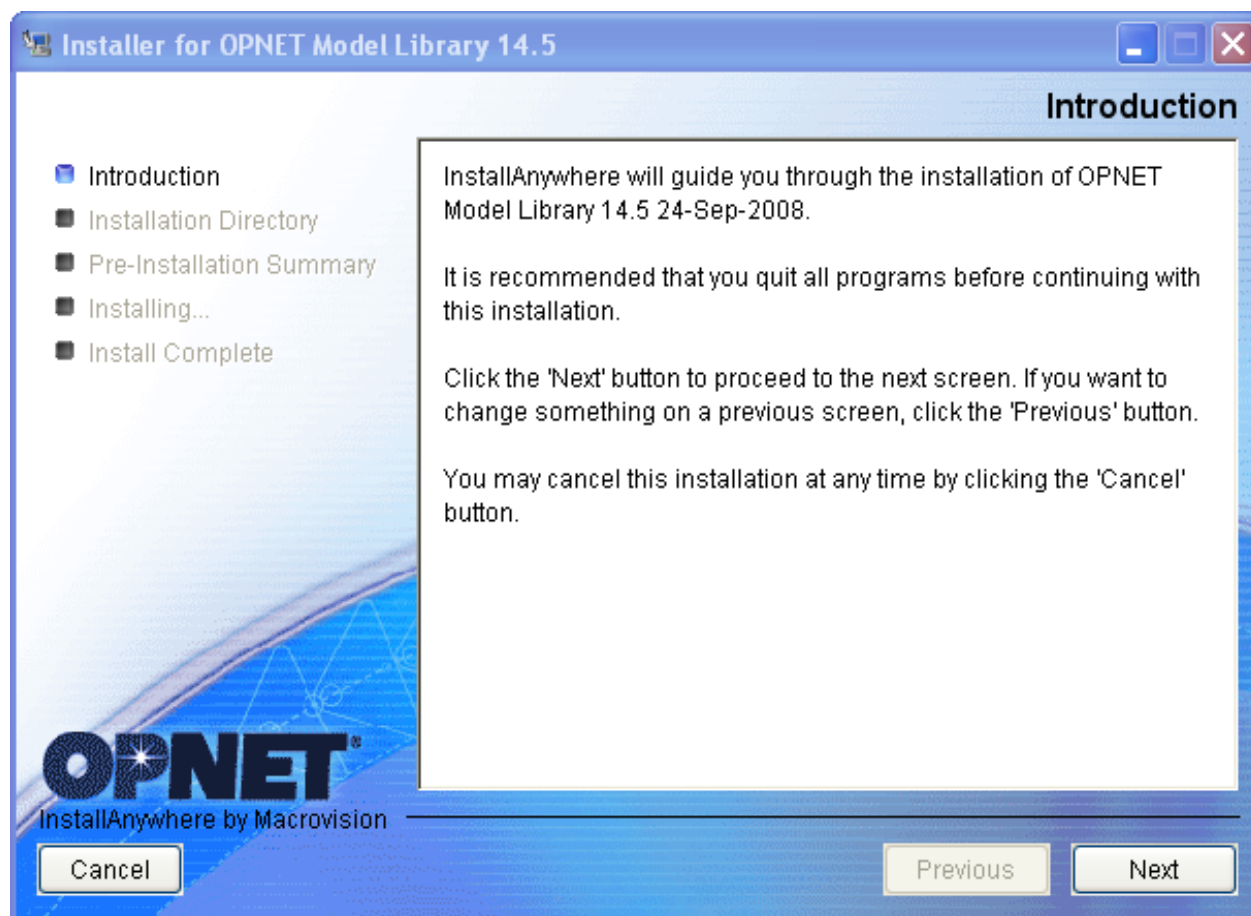
- Opnet Modeler
- Opnet Model Library
- Opnet Model Documentation

ترتیب نصب دقیقا به بصورت ذکر شده است

لطفا به دقت هر مرحله را جهت نصب و پیکر بندی انجام دهید

۱. نصب Opnet Model Library

گام اول : در اولین صفحه گزینه NEXT را انتخاب می نمائیم . این اطلاعات شامل معرفی مقررات

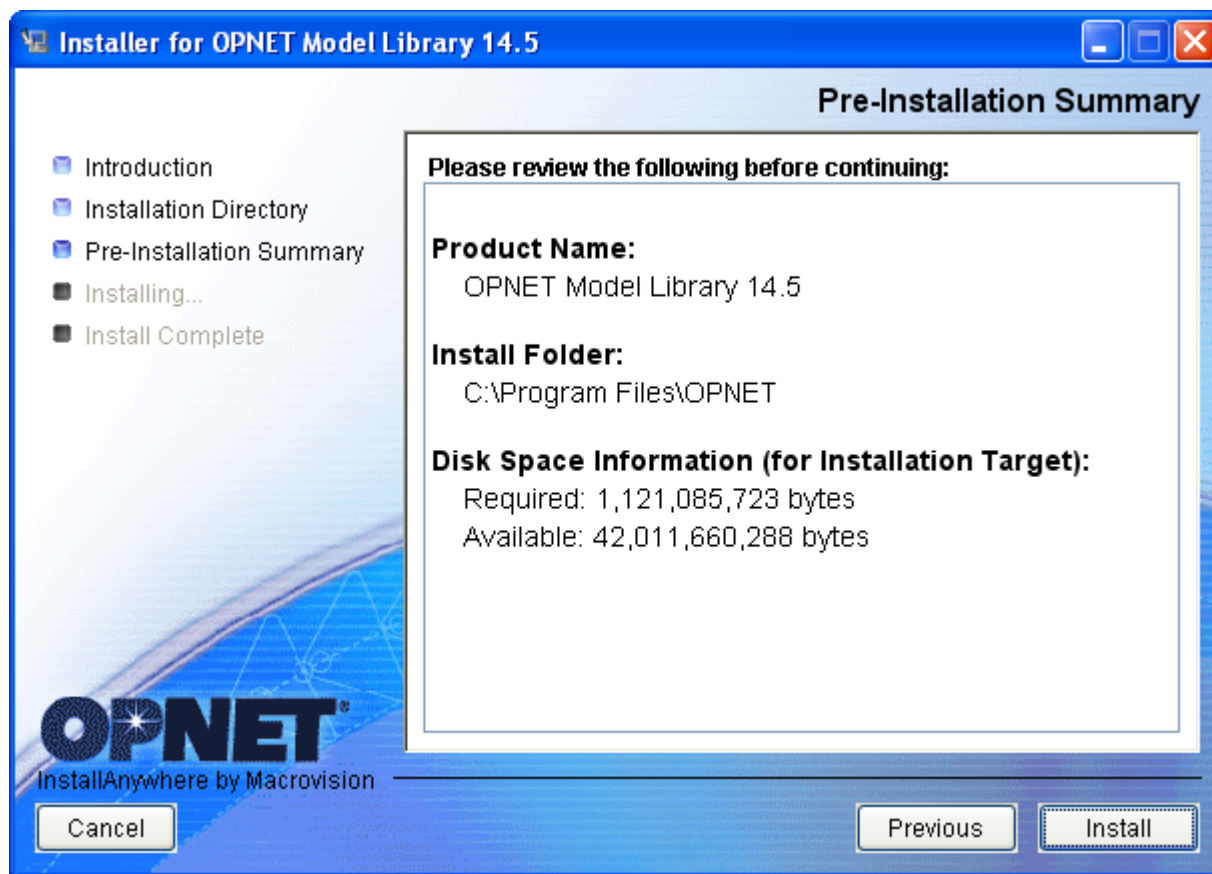


نصب می باشد .

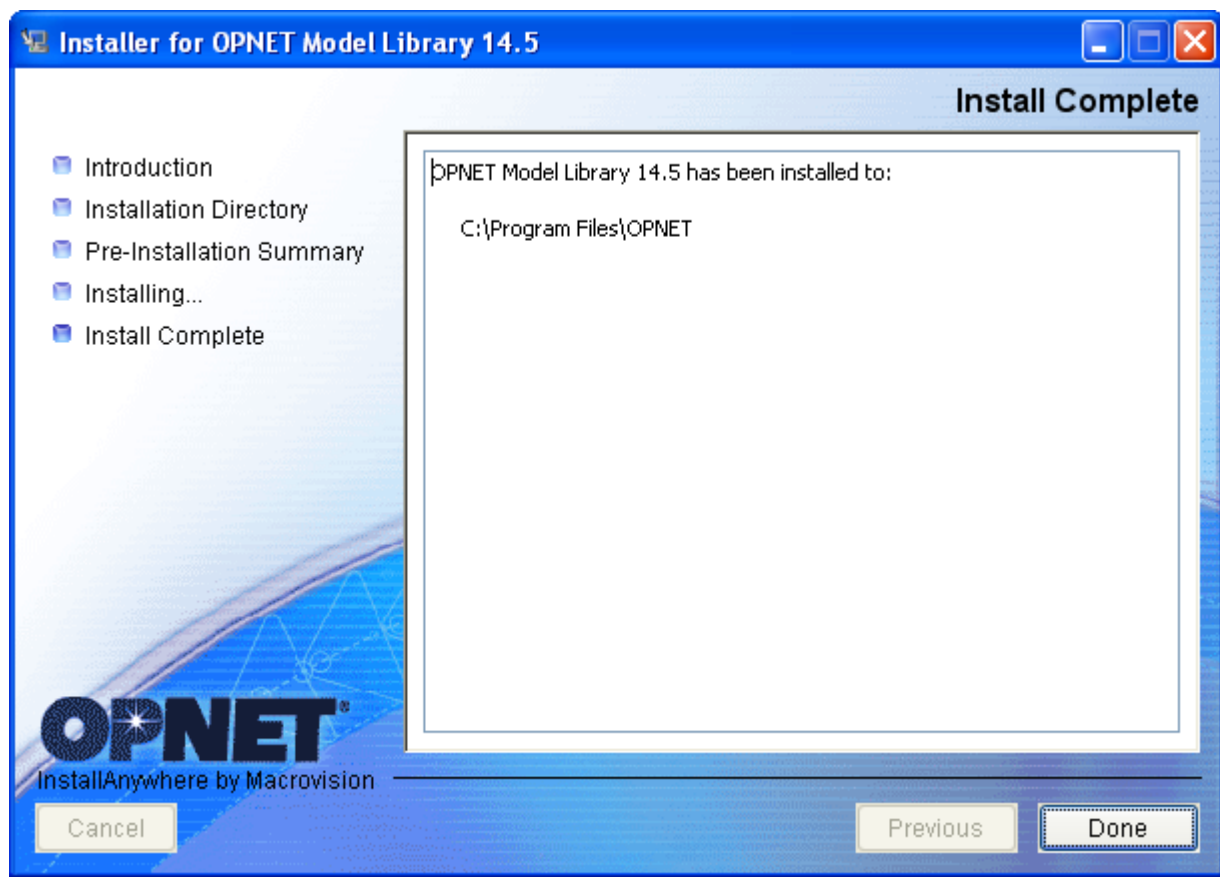
گام دوم : انتخاب یک پوشه در سیستم جهت نصب کتابخانه های نرم افزار



گام سوم : بررسی خلاصه انتخابهای نصب و در صورت صحت فشردن دکمه Install

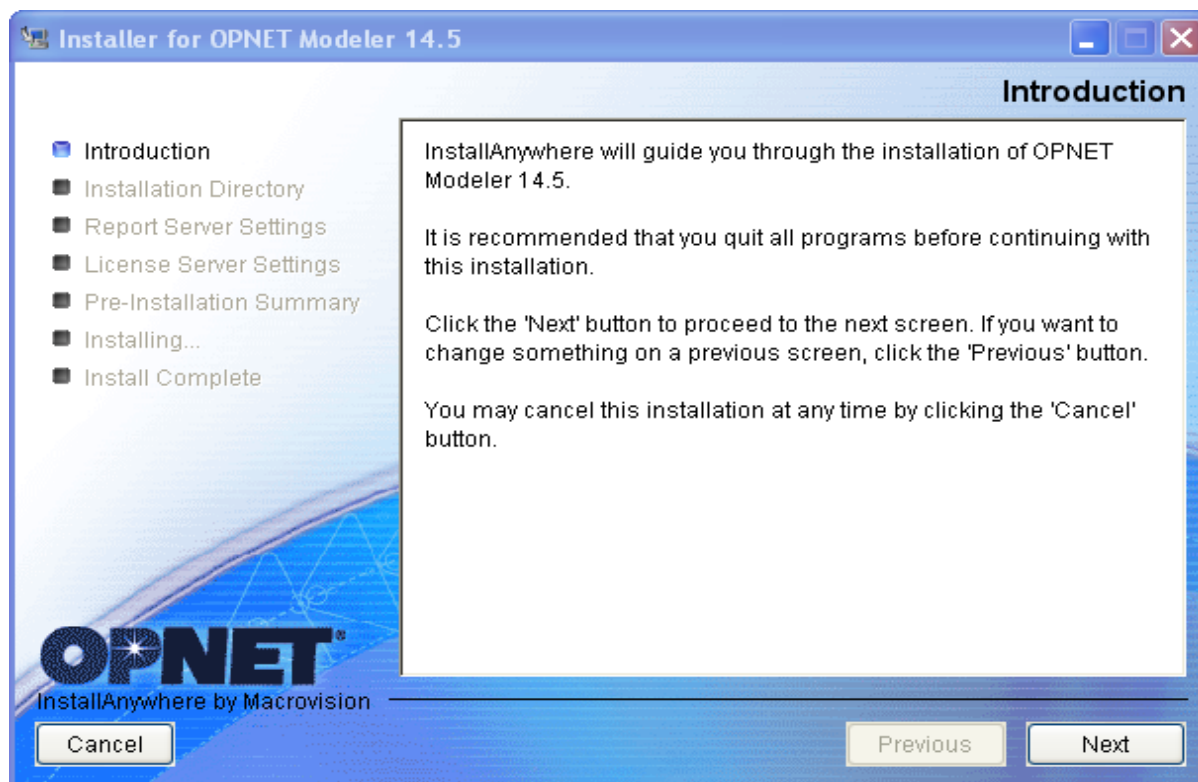


گام چهارم : فشردن کلید Done پس از انجام پروسه نصب



۲. نصب Opnet Modeler

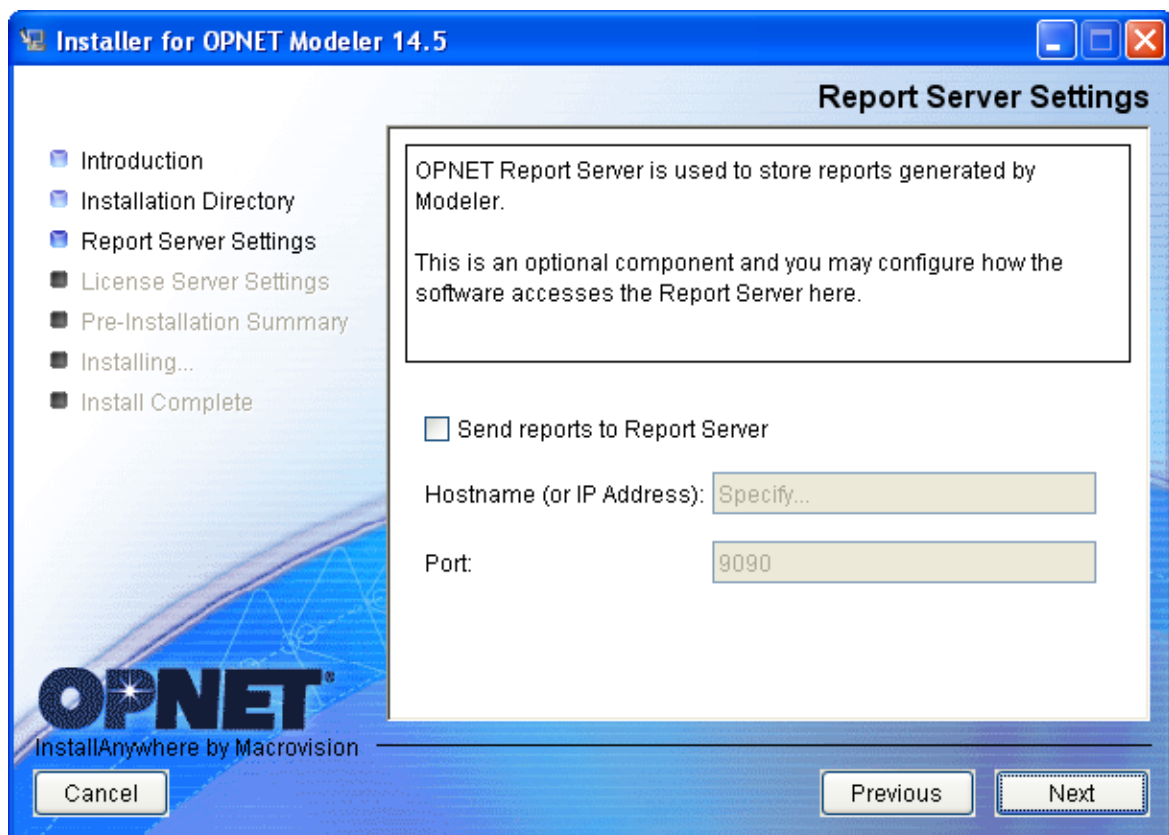
گام اول : در صفحه اول کلید Next را می فشاریم



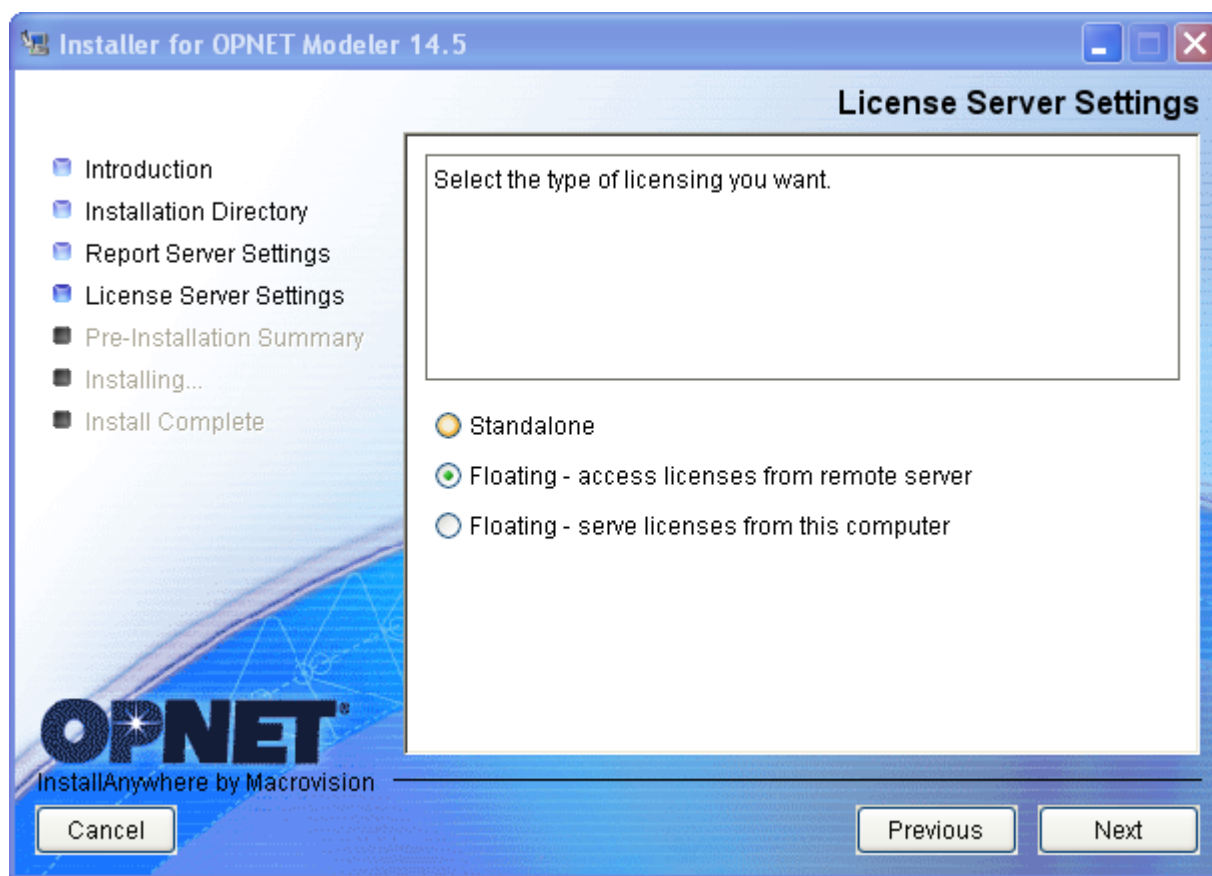
گام دوم : انتخاب یک مسیر جهت نصب modeler در سیستم



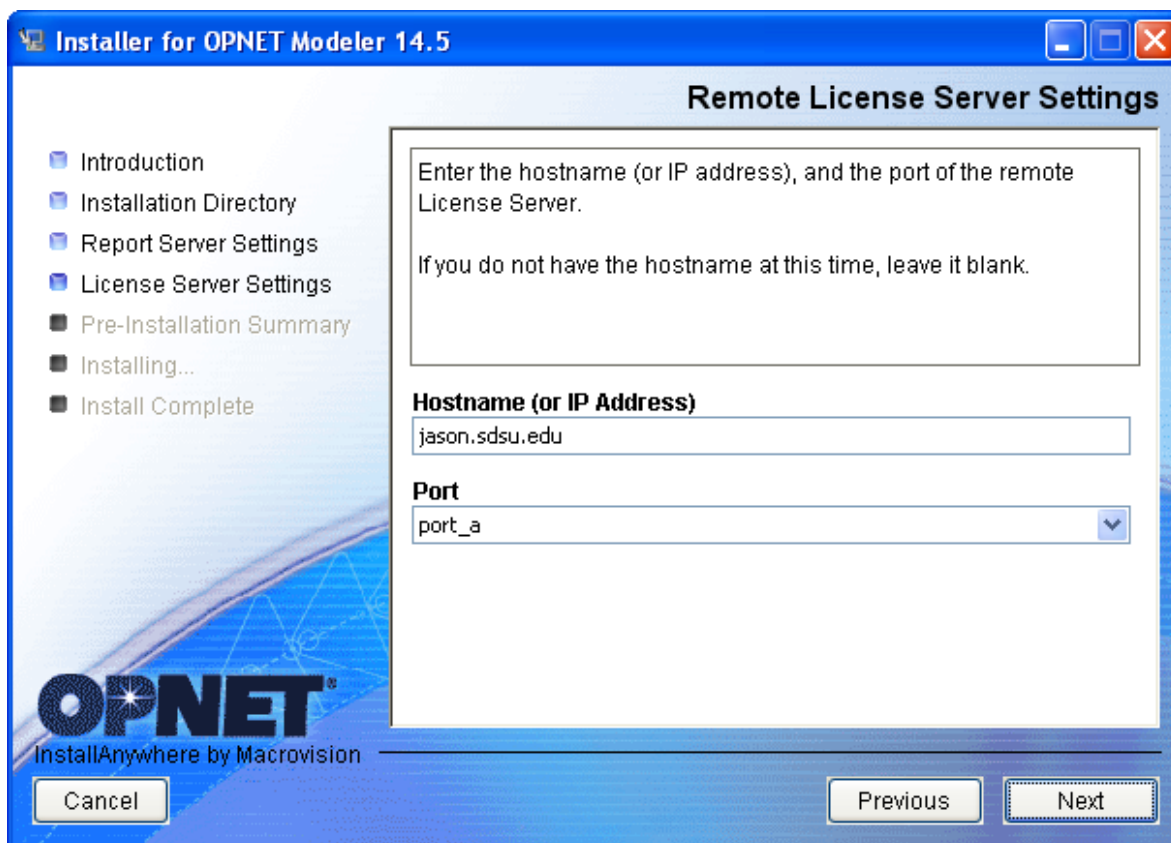
گام سوم: گزینه "Send reports to Report Server" را انتخاب نکنید. کلید Next را بدون تغییری در این صفحه بفشارید



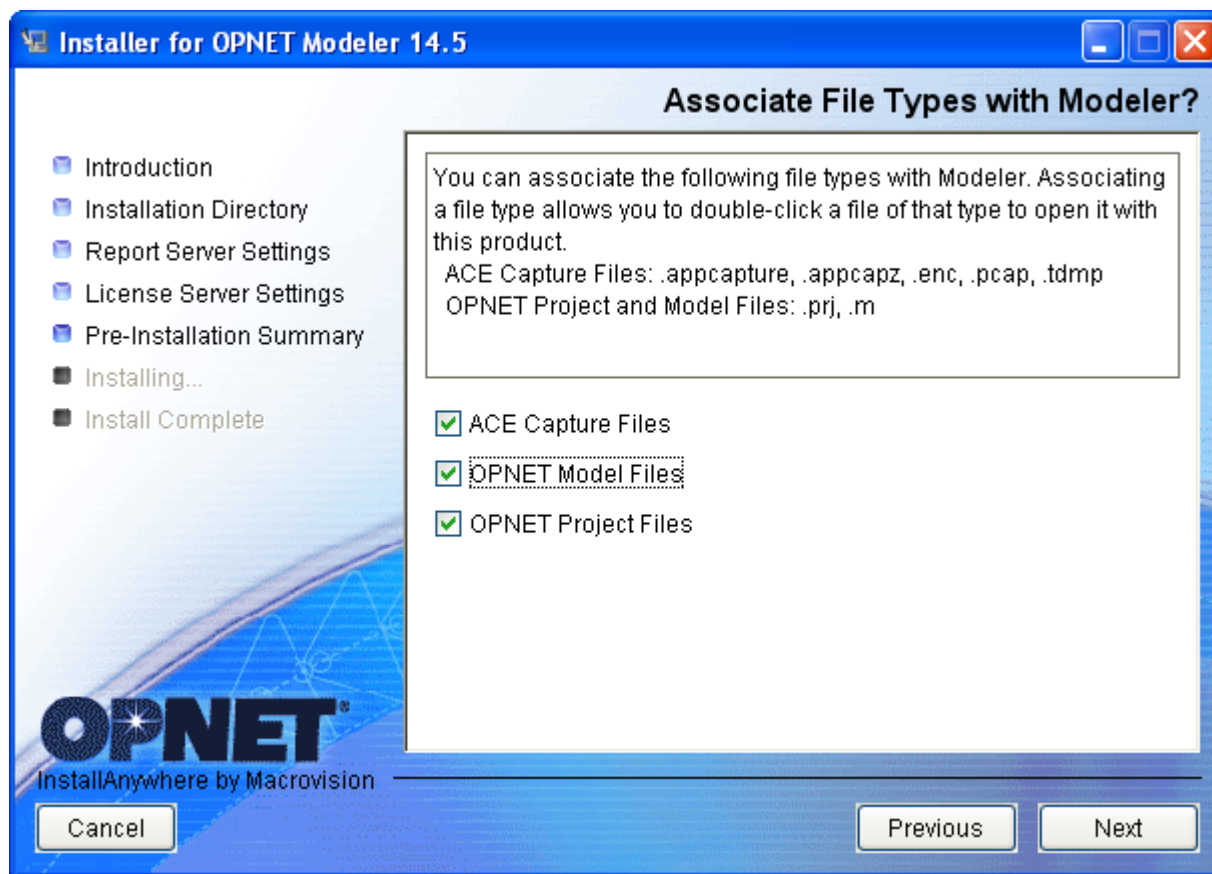
گام چهارم : گزینه "Floating - access license from remote server" را انتخاب و دکمه Next را بفشارید



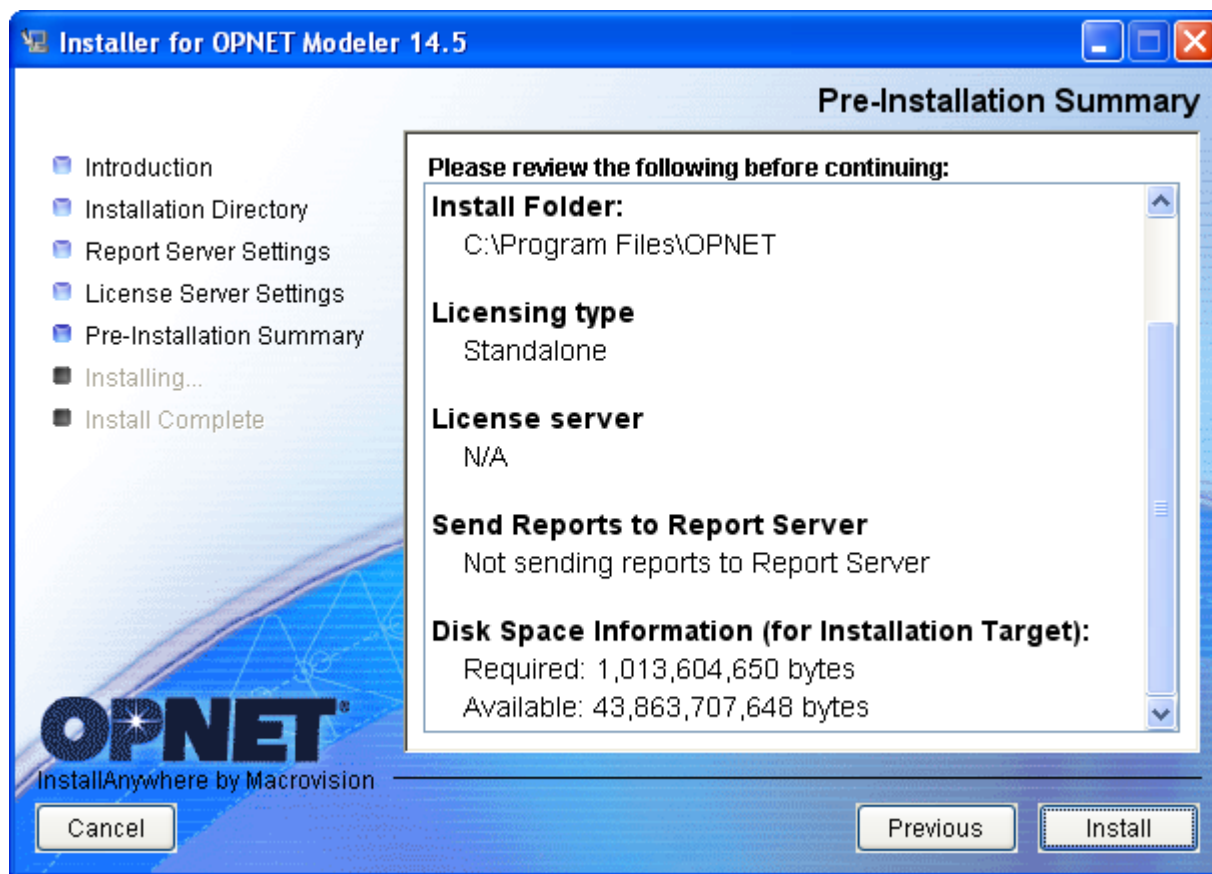
گام پنجم : Hostname سرور license را وارد کرده و کلید Next را بفشارید . این اطلاعات به همراه CD نصب به شما ارائه میگردد .



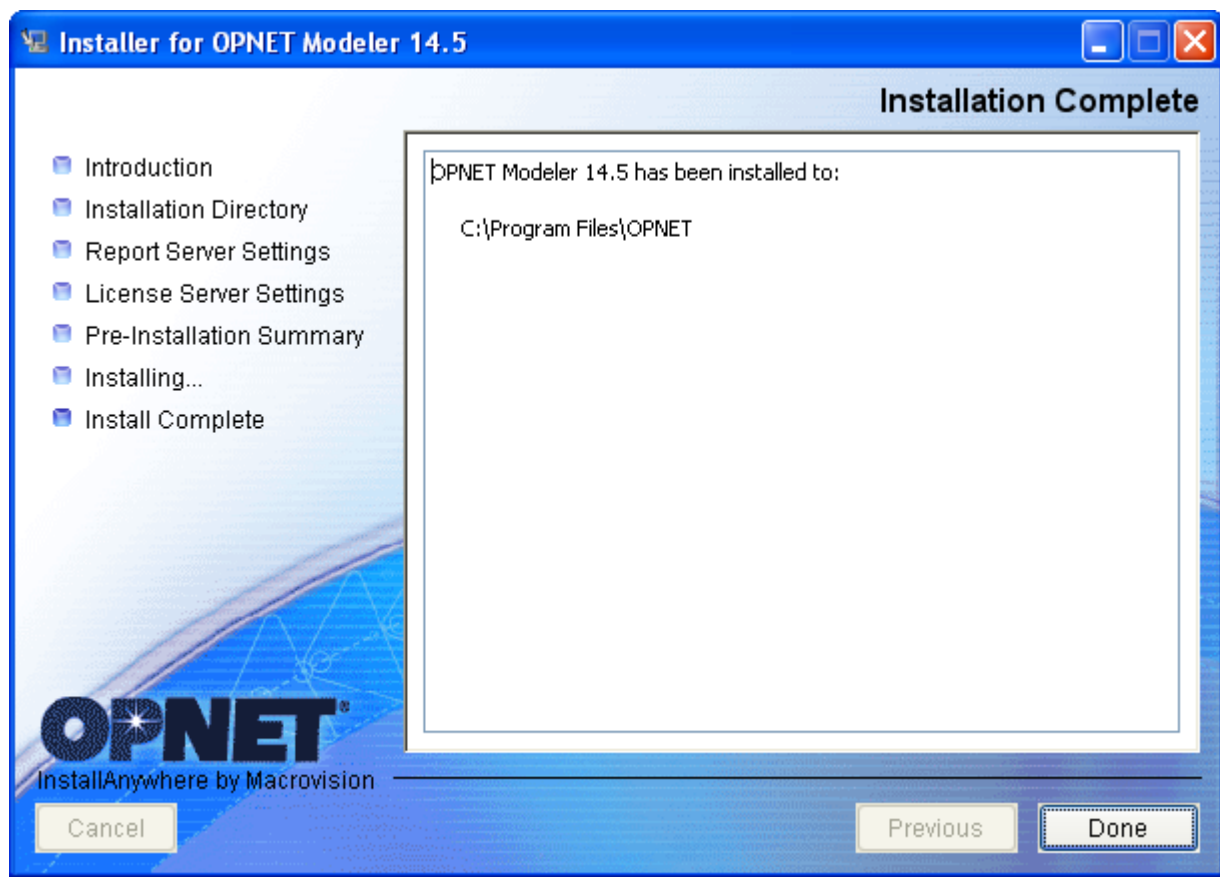
گام ششم: حتما سه گزینه انواع فایل را انتخاب کنید



گام هفتم : خلاصه وضعیت نصب که توسط خود شما تنظیم شده را بررسی و در صورت صحت دکمه Install را بفشارید

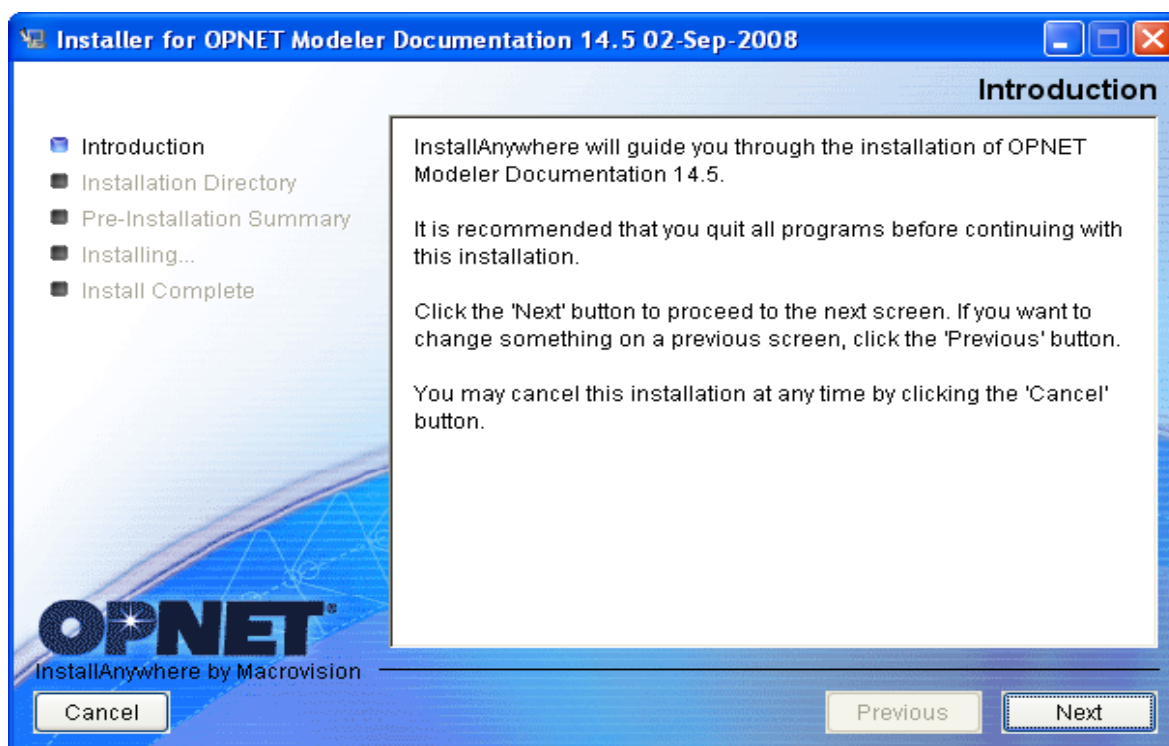


گام هشتم : دکمه Done را جهت خاتمه عملیات نصب بفشارید .

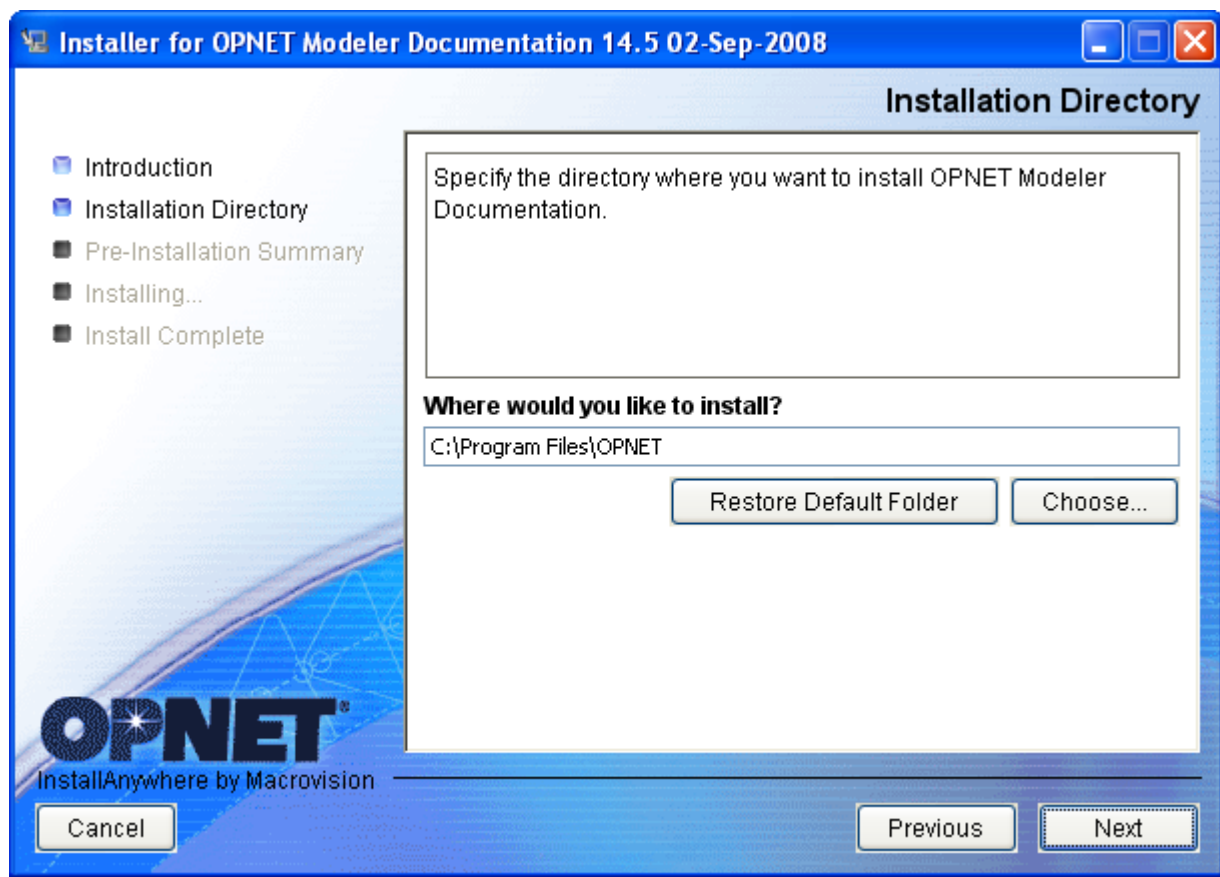


نصب . Opnet Modeler Documentation

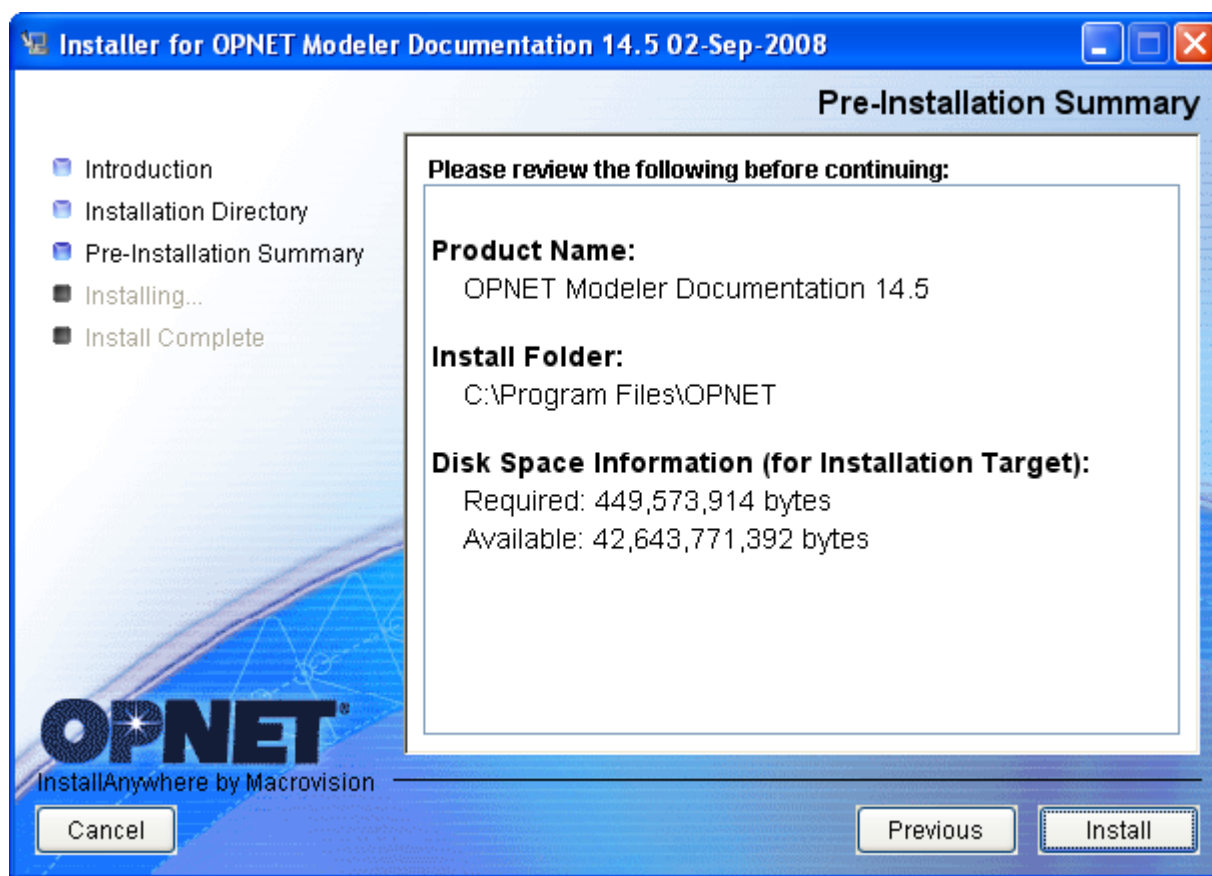
قدم اول : کلید Next را جهت شروع عملیات نصب بفشارید



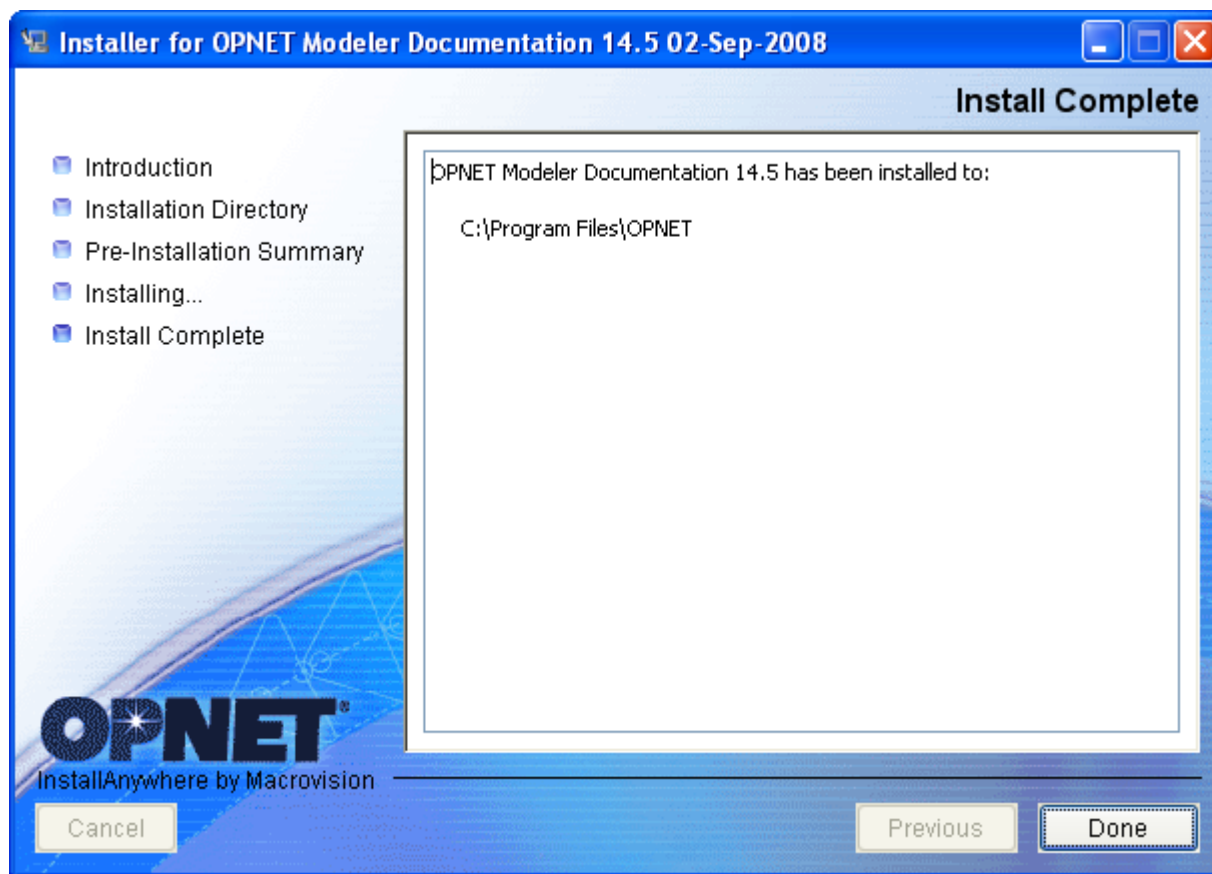
گام دوم: یک پوشه جهت نصب فایل‌های کتابخانه نرم افزار را انتخاب نمایید



گام سوم : خلاصه انتخابهای نصب را مرور کرده و در صورت صحت داده ها نصب را با فشردن دکمه Install آغاز نمایید .



گام چهارم : کلید Done را در آنهای پروسه نصب بفشارید



پس از طی تمام این مرحله لایسنس را نیز نصب نمایید .

گام پنجم :

مسیر های زیر را در سیستم Path نمایید و از صحت آن نیز اطمینان حاصل فرمائید

DevEnvDir = C:\Program Files\Microsoft Visual Studio ۱۰.۰\Common\IDE

FrameworkDir = C: \ WINDOWS \ Microsoft.NET \ Framework

INCLUDE =

C: \ Program Files \ Microsoft Visual Studio ۹.۰ \ VC \ ATLMFC \ INCLUDE;

C: \ Program Files \ Microsoft Visual Studio ۹.۰ \ VC \ INCLUDE;

C: \ Program Files \ Microsoft SDKs \ Windows \ v۶.۰A \ include;

LIB =

C: \ Program Files \ Microsoft Visual Studio 4.0 \ VC \ ATLMFC \ LIB;

C: \ Program Files \ Microsoft Visual Studio 4.0 \ VC \ LIB;

C: \ Program Files \ Microsoft SDKs \ Windows \ v6.0.A \ lib;

LIBPATH =

C: \ WINDOWS \ Microsoft.NET \ Framework \ v2.0;

C: \ WINDOWS \ Microsoft.NET \ Framework \ v2.0.50727;

C: \ Program Files \ Microsoft Visual Studio 4.0 \ VC \ ATLMFC \ LIB;

C: \ Program Files \ Microsoft Visual Studio 4.0 \ VC \ LIB;

PATH =

C: \ Program Files \ Microsoft Visual Studio 4.0 \ Common\ \ IDE;

C: \ Program Files \ Microsoft Visual Studio 4.0 \ VC \ BIN;

C: \ Program Files \ Microsoft Visual Studio 4.0 \ Common\ \ Tools;

C: \ WINDOWS \ Microsoft.NET \ Framework \ v2.0;

C: \ WINDOWS \ Microsoft.NET \ Framework \ v2.0.50727;

C: \ Program Files \ Microsoft Visual Studio 4.0 \ VC \ VCPackages;

C: \ Program Files \ Microsoft SDKs \ Windows \ v6.0.A \ bin;

VCINSTALLDIR =

C: \ Program Files \ Microsoft Visual Studio 4.0 \ VC

VS4.0.COMNTOOLS =

C: \ Program Files \ Microsoft Visual Studio 4.0 \ Common\ \ Tools \

VSINSTALLDIR =

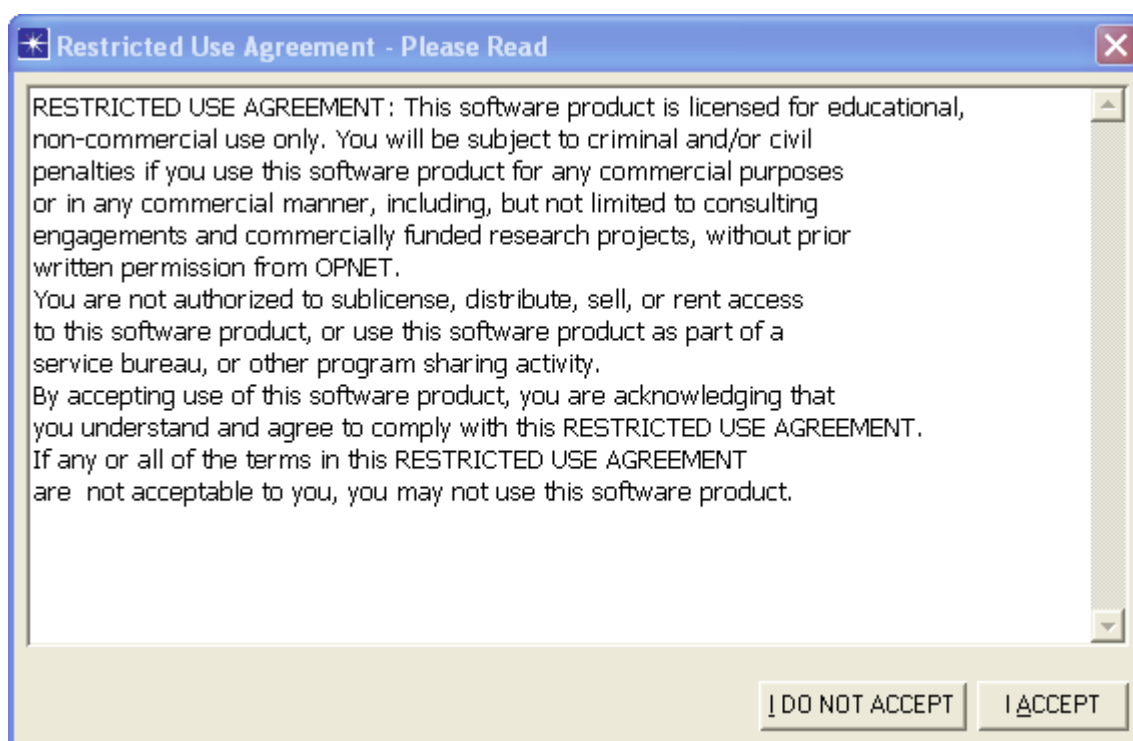
C: \ Program Files \ Microsoft Visual Studio 4.0

WindowsSdkDir =

C: \ Program Files \ Microsoft SDKs \ Windows \ v۶.۰A \

در صورت نصب صحیح سه ماژول فوق و تعریف صحیح مسیرها اکنون میتوانید برنامه را از منوی Start و زیر منوی Programs اجرا نمایید

در ابتدای نصب برنامه در قسمت Accept the user agreement گزینه "I Accept" را انتخاب نمایید .



اکنون شبیه ساز Opnet آماده بهره برداری است

