



به نام خدا
فاز صفر پروژه برنامه نویسی شی گرا
دکتر هاشمی ، دکتر آراسته ، دکتر شیرعلی

مسئولین فاز صفر:

سیدعلی سجادی

@salisajjadii

محمد مهدی شکرزاده

@Mahdi_Shokrzadeh

نیمسال دوم ۱۴۰۳-۰۴



۱ مقدمه

در حوزه‌ی مهندسی برق و الکترونیک، تحلیل و شبیه‌سازی مدارهای الکتریکی نقش حیاتی در طراحی و ارزیابی سیستم‌های پیچیده دارد. شبیه‌سازی مدارها به مهندسان این امکان را می‌دهد که بدون نیاز به پیاده‌سازی فیزیکی، رفتار مدار را در شرایط مختلف بررسی کرده و مشکلات احتمالی را قبل از ساخت تشخیص دهند. با توجه به اهمیت این موضوع، نرم‌افزارهای متعددی برای شبیه‌سازی مدار توسعه یافته‌اند که از میان آن‌ها می‌توان به pSpice و hSpice اشاره کرد. این نرم‌افزارها قادرند با دریافت مشخصات مدار به صورت متنی یا گرافیکی، مدار را از طریق روش‌های عددی حل کرده و پاسخ‌های گذرا و دائمی آن را محاسبه و نمایش دهند.

شبیه‌سازی مدار است که در دهه‌ی ۱۹۷۰ توسعه یافت و همچنان یکی از استانداردهای اصلی در این حوزه محسوب می‌شود. این نرم‌افزار به مهندسان کمک می‌کند که مدارهای آنالوگ و دیجیتال را قبل از پیاده‌سازی واقعی، تحت شرایط مختلف تست کرده و تحلیل کنند.

۲ هدف پروژه

در این پروژه، هدف طراحی و پیاده‌سازی یک شبیه‌ساز مدار به روش برنامه‌نویسی شی‌گرا (OOP) است. مدارهای الکتریکی از مجموعه‌ای از گره‌ها (Nodes) و عناصر (Elements) تشکیل شده‌اند که به روش‌های مختلف به یکدیگر متصل می‌شوند. هر عنصر دارای ویژگی‌های خاصی است که نحوه‌ی عملکرد آن را در مدار تعیین می‌کند. در نرم‌افزارهایی نظیر Spice، توصیف مدار به صورت متنی انجام می‌شود، به این صورت که هر عنصر مدار در قالب یک دستور مشخص شده و شماره‌ی گره‌های متصل به آن، به همراه مقادیر مربوطه، تعیین می‌شوند.

۳ روش پیاده‌سازی

برای توسعه‌ی این شبیه‌ساز، ابتدا ساختار کلی مدار به صورت شی‌گرا طراحی شده و کلاس‌های مختلفی برای مدل‌سازی اجزای مدار ایجاد می‌شوند. این کلاس‌ها شامل موارد زیر هستند:

- کلاس مدار (Circuit): مسئول نگهداری و مدیریت تمامی عناصر مدار و گره‌ها
- کلاس گره (Node): نشان‌دهنده‌ی نقاط اتصال عناصر مدار
- کلاس عنصر (Element): شامل قطعاتی مانند مقاومت، خازن، سلف و منابع تغذیه، که ویژگی‌ها و رفتارهای مشخصی دارند



روابط بین این کلاس‌ها با استفاده از اصول برنامه‌نویسی شی‌گرا مانند وراثت (Inheritance) ترکیب (Com-position) و وابستگی (Dependency) طراحی شده‌اند. این معماری باعث می‌شود که سیستم انعطاف‌پذیر، قابل توسعه و مقیاس‌پذیر باشد و بتواند در آینده قابلیت‌های جدیدی را پشتیبانی کند.

۴ اهمیت و کاربرد

این پروژه فرصتی برای درک عمیق‌تر مفاهیم برنامه‌نویسی شی‌گرا و طراحی نرم‌افزارهای مهندسی فراهم می‌کند. پیاده‌سازی صحیح این شبیه‌ساز می‌تواند به دانشجویان کمک کند تا مفاهیم مدارهای الکتریکی و روش‌های حل عددی آن‌ها را بهتر درک کنند. همچنین، این پیاده‌سازی می‌تواند پایه‌ای برای توسعه‌ی شبیه‌سازهای پیشرفته‌تر با قابلیت‌های تحلیل گذرا، تحلیل حالت پایدار، پشتیبانی از ورودی‌های گرافیکی و خروجی‌های تصویری باشد.



۵ پرسش‌های اجباری

۱.۵ طراحی کلاس‌ها و ساختار برنامه

۱.۱.۵ پرسش ۱:

با توجه به مفهوم اشیاء و کلاس‌ها، کلاس‌های اصلی مورد نیاز برای پیاده‌سازی شبیه‌ساز SPICE را شناسایی کنید. برای هر کلاس، ویژگی‌ها (متغیرهای داده‌ای) و رفتارها (متدها) را مشخص کنید. حداقل کلاس‌های زیر را در نظر بگیرید:

- کلاس Circuit (مدار)

- کلاس Node (گره)

- کلاس Element (عنصر مدار)

- کلاس‌های مشتق شده از Element برای عناصر مختلف مدار (مقاومت، خازن، سلف، منابع و غیره)

توضیح: در این پرسش، هدف تشخیص داده‌ها و عملیات هر کلاس است. برای مثال، کلاس مدار باید بتواند المان‌ها و گره‌ها را نگهداری کند و عملیاتی مانند اضافه کردن المان، حذف المان و تحلیل مدار را انجام دهد.

۲.۱.۵ پرسش ۲:

نمودار UML کلاس‌ها را برای سیستم شبیه‌ساز SPICE طراحی کنید. روابط بین کلاس‌ها را با استفاده از مفاهیم وراثت (inheritance)، ترکیب (composition) و وابستگی (dependency) نشان دهید. توضیح: نمودار UML باید نشان دهد که چگونه کلاس‌ها با یکدیگر ارتباط دارند. به طور مثال، یک مدار از مجموعه‌ای از گره‌ها و المان‌ها تشکیل شده است (ترکیب) و کلاس‌های مقاومت، خازن و غیره از کلاس Element مشتق می‌شوند (وراثت).

۲.۵ پیاده‌سازی کلاس‌ها و روابط آن‌ها

۱.۲.۵ پرسش ۳:

برای کلاس Element یک class abstract طراحی کنید که متدهای مورد نیاز برای همه عناصر مدار را تعریف کند. سپس حداقل ۵ کلاس مشتق شده از Element را برای عناصر مختلف مدار (مانند مقاومت، خازن، سلف، منبع ولتاژ و منبع جریان) طراحی کنید. برای هر کدام، متدهای مجازی را در نظر بگیرید (نیازی به پیاده‌سازی کامل نیست صرفاً نام بردن کلی متدها و توضیح عملکرد هر کدام کافی است).



۲.۲.۵ پرسش ۴:

با توجه به مفهوم چندریختی (polymorphism) در برنامه نویسی شیءگرا، توضیح دهید که چگونه می توان از این مفهوم برای مدیریت عناصر مختلف مدار استفاده کرد. مثالی از کد را ارائه دهید که نشان دهد چگونه می توان از یک آرایه یا وکتور از اشاره گرهای Element برای نگهداری و مدیریت عناصر مختلف مدار استفاده کرد.

توضیح: چندریختی به ما اجازه می دهد که با عناصر مختلف مدار به صورت یکسان رفتار کنیم. به عنوان مثال، می توانیم یک وکتور از اشاره گرهای پایه Element داشته باشیم و انواع مختلف عناصر را در آن ذخیره کنیم و بدون نیاز به بررسی نوع هر عنصر، متدهای مشترک را روی آنها صدا بزنیم.

۳.۵ الگوهای طراحی و تحلیل مدار

۱.۳.۵ پرسش ۵:

یک pattern design مناسب برای خواندن و تفسیر فایل های توصیف مدار SPICE پیشنهاد دهید. توضیح دهید که چرا این الگو مناسب است و چگونه آن را پیاده سازی خواهید کرد.

توضیح: در اینجا می توانید از الگوهایی مانند Method Factory Interpreter یا Builder استفاده کنید. برای مثال، الگوی Method Factory می تواند برای ایجاد انواع مختلف عناصر مدار بر اساس خط های فایل ورودی استفاده شود.

۲.۳.۵ پرسش ۶:

در مورد روش هایی عددی برای حل مدار های الکتریکی تحقیق کرده و چند مورد را توضیح دهید. در مورد الگوریتم های حل معادلات و دستگاه معادلات خطی تحقیق کنید. به صورت خاص در مورد روش هایی مانند Elimination Gaussian یا تجزیه LU تحقیق کنید.

۳.۳.۵ پرسش ۷:

یکی از چالش های شبیه سازی مدار، حل دستگاه معادلات خطی بزرگ است. توضیح دهید که برای حل این مشکل چه خواهید کرد. آیا از کتابخانه های خارجی استفاده می کنید یا الگوریتم های خود را پیاده سازی می کنید؟

توضیح: در اینجا می توانید از الگوی Bridge برای جدا کردن واسط حل معادلات از پیاده سازی آن استفاده کنید. این به شما اجازه می دهد که به راحتی بین پیاده سازی های مختلف (مانند کتابخانه های مختلف یا الگوریتم های مختلف) سوییچ کنید.

۴.۳.۵ پرسش ۸:

چگونه می توان پاسخ گذرا و پاسخ دائمی یک مدار را تحلیل کرد؟ با توجه به لزوم پیاده سازی مدار های مرتبه یک و دو در مورد روش های حل دستگاه معادلات دیفرانسیل تحقیق کنید. روش های Euler یا Runge-Kutta را به صورت مختصر توضیح دهید.



۵.۳.۵ پرسش ۹:

برای پیاده‌سازی الگوریتم‌های نمایش نمودارهای مدار (تحلیل، DC تحلیل AC)، از چه طراحی استفاده خواهید کرد؟ توضیح دهید و دلایل خود را بیان کنید.

۴.۵ مدیریت خطا و استثناءها

۱.۴.۵ پرسش ۱۰:

طراحی مناسبی برای مدیریت خطاها و استثناءهای احتمالی در شبیه‌ساز SPICE ارائه دهید. چه نوع استثناهایی ممکن است در هنگام parse کردن فایل ورودی، ساخت مدار یا تحلیل مدار رخ دهد؟ چگونه آن‌ها را مدیریت می‌کنید؟ توضیح: در این سوال باید سلسله مراتب استثناءهای خود را طراحی کنید. برای مثال، می‌توانید یک کلاس پایه SPICE-Exception تعریف کنید و سپس کلاس‌های مشتق شده مانند CircuitException ParsingException و Analy-sisException را از آن مشتق کنید. همچنین باید مشخص کنید که در کجای برنامه این استثناءها پرتاب و در کجا گرفته می‌شوند. نیازی به پیاده‌سازی دقیق و کامل نیست صرفاً بیان ساختار کلی کفایت می‌کند.

۵.۵ پیاده‌سازی رابط کاربری

۱.۵.۵ پرسش ۱۱:

طراحی خود را برای پیاده‌سازی رابط کاربری کنسولی (CLI) شبیه‌ساز SPICE توضیح دهید. چگونه از الگوهای طراحی مناسب مانند Command برای پیاده‌سازی دستورات کاربر استفاده می‌کنید؟ آیا از رویکرد لایه‌بندی برای جدا کردن منطق شبیه‌سازی از رابط کاربری استفاده می‌کنید؟ توضیح: در این سوال، شما باید نحوه طراحی رابط کاربری کنسولی را تشریح کنید. برای مثال، می‌توانید از الگوی Command برای پیاده‌سازی دستورات مختلف کاربر مانند load، show run و غیره استفاده کنید. همچنین می‌توانید از الگوی MVC یا MVP برای جداسازی لایه‌های برنامه استفاده کنید.



۶ پرسش‌های امتیازی

۱.۶ پرسش امتیازی ۱:

توضیح دهید که چگونه می‌توان از Pattern Design Factory برای ایجاد عناصر مدار مختلف بر اساس ورودی کاربر استفاده کرد. یک نمونه کد را ارائه دهید (نیازی به پیاده سازی کامل نیست).
توضیح: الگوی Factory به ما اجازه می‌دهد که ایجاد اشیاء را به کلاس دیگری واگذار کنیم. در اینجا می‌توانید یک کلاس ElementFactory طراحی کنید که بر اساس خط ورودی، SPICE عنصر مناسب را ایجاد می‌کند.

۲.۶ پرسش امتیازی ۲:

بررسی کنید که چگونه می‌توان از Pattern Design Observer برای نمایش و به‌روزرسانی نتایج شبیه‌سازی در زمان واقعی استفاده کرد.
توضیح: الگوی Observer به ما اجازه می‌دهد که وقتی وضعیت یک شیء (Subject) تغییر می‌کند، همه اشیاء وابسته به آن (Observers) به طور خودکار مطلع و به‌روزرسانی شوند. در اینجا، مدار می‌تواند به عنوان Subject و عناصر نمایشی GUI به عنوان Observers عمل کنند.

۳.۶ پرسش امتیازی ۳:

یک طراحی برای سیستم ذخیره‌سازی و بارگذاری مدارها با استفاده از مفاهیم سریالی‌سازی (Serialization) و دسریالی‌سازی (Deserialization) ارائه دهید.
توضیح: در این سوال، شما باید نحوه ذخیره و بازیابی مدارها را تشریح کنید. می‌توانید از الگوی Memento برای ذخیره وضعیت داخلی مدار استفاده کنید یا از تکنیک‌های سریالی‌سازی ++C مانند استفاده از «operator» و «operator» برای ذخیره و بازیابی وضعیت استفاده کنید.

۴.۶ پرسش امتیازی ۴:

طراحی یک سیستم پلاگین برای شبیه‌ساز SPICE را توضیح دهید که به کاربران اجازه دهد المان‌های سفارشی خود را بدون نیاز به تغییر کد اصلی برنامه اضافه کنند.
توضیح: در این سوال، شما باید یک سیستم پلاگین طراحی کنید که به کاربران امکان دهد المان‌های سفارشی خود را به صورت کتابخانه‌های پویا پیاده‌سازی کرده و به برنامه اضافه کنند. می‌توانید از الگوهایی مانند Factory Abstract و Dependency Injection برای این منظور استفاده کنید.



۵.۶ پرسش امتیازی ۵:

طراحی یک سیستم ثبت وقایع یا همان System Logging برای شبیه ساز SPICE را توضیح دهید. چگونه از الگوهای طراحی مانند Singleton و Decorator برای پیاده سازی یک سیستم ثبت وقایع قابل تنظیم و کارآمد استفاده می کنید؟ توضیح: سیستم ثبت وقایع برای اشکال زدایی و پایش عملکرد شبیه ساز بسیار مهم است. می توانید از الگوی Singleton برای ایجاد یک نمونه واحد از Logger استفاده کنید و از الگوی Decorator برای افزودن قابلیت های مختلف به Logger مانند ثبت در فایل، نمایش در کنسول و غیره استفاده کنید.

۶.۶ پرسش امتیازی ۶:

در مورد نحوه ذخیره سازی داده های مدار و همچنین داده های شبیه سازی تحقیق کنید. به چه روشی میتوان این داده ها را ذخیره کرد تا با بستن و باز کردن مجدد برنامه بتوان به مدار های قبلی دسترسی پیدا کرد. به طور عمده بر روی داده ساختار های sql و Json تحقیق کنید. بیان کنید هر داده با چه تاییبی ذخیره خواهد شد.



۷ ارزیابی و تحویل

- محتویات تحویلی:

- گزارش پروژه به صورت PDF شامل:

- توضیح طراحی کلی پروژه
- نمودار UML کلاس‌ها
- توضیح الگوریتم‌های پیاده‌سازی شده و پاسخ سوالات
- نتایج بدست آمده (در صورت وجود!)

- معیارهای ارزیابی:

- طراحی شی‌گرا و استفاده از اصول و الگوهای مناسب (۵۰٪)
- گزارش پروژه (۵۰٪)
- تحویل در قالب لتک (۱۰٪ امتیاز اضافی)
- پرسش‌های امتیازی (تا ۳۰٪ امتیاز اضافی)