

لطفاً دو سوال از سوال‌های داده شده را به انتخاب خود حل کنید. در صورتی که می‌خواهید از نمره‌ی ارفاق استفاده کنید می‌توانید یک سوال اضافی‌تر (در مجموع سه سوال) حل کنید ولی بیش از سه سوال را پاسخ ندهید.

۱- در یک سیستم از روش TMR برای Mask کردن اشکال‌ها استفاده شده است و در عین حال سیستم دارای یک پدک گرم برای هر واحدی است که خراب شود و سپس یک پدک سرد برای واحد بعدی است که خراب شود. روش تشخیص خرابی واحدها به این شکل است که خروجی آن‌ها با حاصل رأی‌گیری مقایسه می‌شود. این سیستم بیش از این دو پدک واحد پدک دیگری ندارد. مقدار TTF برای تمام اجزاء از توزیع نمایی با نرخ μ پیروی می‌کند، غیر از واحد رأی‌گیر و مقایسه‌کننده‌ها که آن‌ها نیز دارای TTF با توزیع نمایی ولی با نرخ μ هستند. در این سیستم اگر رأی‌گیر خراب شود سیستم به شکل فاجعه‌آمیز خراب خواهد شد اما اگر رأی‌گیر سالم باشد ولی واحدهای خراب به تعدادی باشند که سیستم دیگر نتواند به فعالیت ادامه دهد سیستم به شکل ایمن خاموش می‌شود. برای این سیستم $S(t)$ را با استفاده از مدل مارکوف به دست آورید. دقت کنید که با استفاده از نرم‌افزار maple می‌توانید دستگاه معادلات دیفرانسیل را حل کنید.

۲- در یک سیستم بی‌درنگ یک وظیفه دارای زمان اجرا با توزیع یکنواخت است که WCET آن برابر با 10ms و بهترین زمان اجرای آن برابر 1ms است و دارای Deadline برابر با 20ms است. سیستم از روش Checkpointing برای تحمل‌پذیری اشکال استفاده می‌کند. برای فواصل Checkpointها دو سناریو را در نظر می‌گیریم:

الف) پنج Checkpoint به فواصل یکسان داریم که طول هر بخش 2ms است.

ب) پنج Checkpoint داریم به فواصل غیر یکسان که طول آن‌ها به ترتیب از اولین بخش به آخرین بخش عبارت است از: 1ms، 1.5ms، 2ms، 2.5ms و 3ms.

از سربار Checkpointها صرف نظر شده است. هنگامی که قطعه‌ای از برنامه زمان اجرایی برابر L (برحسب ms) دارد احتمال رخداد اشکال در آن برابر $10^{-4} * L$ است. همچنین هنگامی که قطعه‌ای از برنامه زمان اجرایی به طول L (برحسب ms) دارد مقدار $100 * L$ انرژی مصرف می‌کند. با استفاده از یک قطعه کد (زبان‌های C، Java و Python مجاز است) شبیه‌سازی مونت کارلو انجام دهید و حساب کنید که برای دو سیستم الف و ب، احتمال خرابی چقدر است و مقدار انرژی متوسط مصرفی چقدر است.

۳- یک بانک ثبات داریم با هشت عدد ثبات ۱۶ بیتی که دارای یک درگاه نوشتن و دو درگاه خواندن است. برای تحمل‌پذیری اشکال دو کپی از این بانک ثبات مورد استفاده است که یکی واحد اصلی و دیگری پشتیبان است. اطلاعات یکسانی در هر دو نوشته می‌شود. اما موقع خواندن همیشه از واحد اصلی اطلاعات را می‌خوانیم مگر اینکه در واحد اصلی خطا کشف شود که در این صورت از پشتیبان اطلاعات را دریافت می‌کنیم. واحد پشتیبان دارای مکانیسم کشف خطا نیست. واحد اصلی از مکانیسم کشف خطای Checksum استفاده می‌کند. به این ترتیب که حال جمع تمام ثبات‌های ۱۶ بیتی همیشه در یک ثبات ۱۶ بیتی ذخیره می‌شود. موقع خواندن همیشه Checksum بررسی می‌شود و اگر همخوانی وجود نداشت خطا کشف شده است.

این سیستم را با استفاده از زبان Verilog یا VHDL توصیف کنید. به نحوی که قابلیت تزریق اشکال در بانک ثبات اصلی و پشتیبان وجود داشته باشد. تزریق اشکال می‌تواند n بیت از میان $2 * 8 * 16 + 19$ بیتی را که داریم خراب (Bit Flip) کند و هرکدام از بیت‌ها ممکن است با شانس مساوی انتخاب شوند. برای چهار حالت $n=1$ ، $n=2$ ، $n=3$ و $n=4$ آزمایش تزریق اشکال را انجام دهید و هر بار هزار اشکال تزریق کنید. برای هرکدام از چهار حالت $n=1$ ، $n=2$ ، $n=3$ و $n=4$ احتمال اینکه سیستم بتواند اشکال را تحمل کند چقدر است؟ در هر آزمایش تزریق اشکال ابتدا یک سری داده‌ی دلخواه در ثبات‌ها بنویسید. بعد تزریق اشکال کنید و بعد داده‌ها را بخوانید و اگر همخوانی وجود نداشت خطا در نظر بگیرید.

۴- یک روش کشف خطا در برنامه‌ها این است که هنگام نوشتن یک برنامه از ویژگی‌های خاص همان برنامه یا الگوریتم در ارائه‌ی روش کشف خطا استفاده کنیم. به چنین روش‌هایی Algorithmic Level Fault Tolerance گفته می‌شود. مثلاً هنگام ضرب دو ماتریس A و B در یکدیگر اگر اولی را در ۳ ضرب کنیم و دومی را در ۵ آنگاه تمام مؤلفه‌های ماتریس حاصل باید تقسیم‌پذیر بر 15 باشد و در غیر اینصورت خطا رخ داده است. این روش تمام خطاهای نوع SBE را کشف می‌کند (چرا؟). اگر اشکال‌های از نوع double bit error را در نظر بگیریم (دو بیت خرابی می‌توانند با احتمال یکسان در هر دو بیت مختلفی باشند) چه کسری از اشکال‌ها توسط روش ذکر شده کشف می‌شود؟ چه روش ALFT را برای پردازش گراف‌هایی پیشنهاد می‌کنید که از لیست‌های پیوندی دو طرفه استفاده می‌کنند؟ با ترسیم شکل روش ALFT خود را شرح دهید.

۵- توصیف یک جمع کننده با زنجیره Manchester که با کد SPICE انجام شده است را در نظر بگیرید. برای مدل کردن اثر برخورد ذرات به مدار در SPICE از منابع جریان متغیر استفاده می‌شود و به این شکل می‌توان آزمایش‌های تزریق اشکال انجام داد (منبع جریان در منابع مربوطه ارائه شده است و در اینترنت موجود است). جمع کننده‌ی ۸ بیتی در فناوری ۹۰ نانومتر را در نظر بگیرید. ذراتی را در نظر بگیرید که برخورد آن‌ها بتواند ولتاژ خازن‌های گیت ترانزیستورها را کامل از صفر منطقی به یک منطقی و یا برعکس سوئیچ کند. حال هزار عدد از این ذرات را به نقاط مختلف جمع کننده‌ی ۸ بیتی با احتمال یکسان اعمال کنید آن هم هنگامی که جمع کننده در حال جمع کردن اعداد تصادفی است. اعداد تصادفی که ورودی جمع کننده هستند نیز می‌توانند با احتمال یکسان هر عددی باشند. اعداد ورودی را می‌توانید با یک برنامه دیگر مثلاً با زبان Java تولید کرده و در یک فایل ذخیره کرده باشید و SPICE فقط از آن فایل ورودی‌ها را بخواند. چه کسری از این اشکال‌های تزریق شده منجر به خطا می‌شود؟

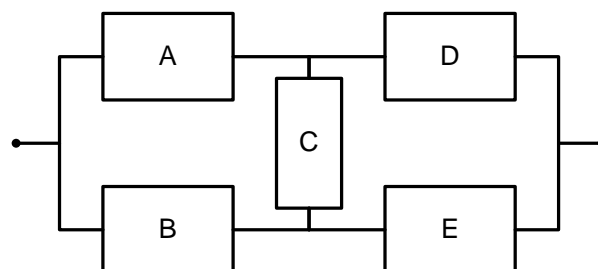
۶- رابطه‌ی TTF یک سیستم با اجزاء آن به شکلی که در ادامه آمده است مشخص می‌شود.

$$T_{Sys} = \max(T_1, T_2) + \min(T_3, T_4)$$

قابلیت اطمینان این سیستم را با مدل مارکوف به دست آورید. نرخ خرابی در هر سه واحد یکسان و برابر با λ است.

۷- یک سیستم متشکل از ۴ واحد کامپیوتری داریم که اگر فقط ۲ عدد از کامپیوترها سالم باشند سیستم سرویس خود را ارائه می‌کند. واحدهای کامپیوتری دارای MTTF برابر ۱۰۰ ساعت هستند و TTF آن‌ها توزیع نمایی دارد. همچنین هر سیستم که خراب می‌شود دارای MTTR برابر ۵۰ ساعت است ولی TTR از توزیع یکنواخت (از صفر تا صد ساعت) پیروی می‌کند. مقدار Steady Availability این سیستم چقدر است؟ برای محاسبه‌ی پاسخ، هم می‌توانید از شبیه‌سازی مونت کارلو استفاده کنید و هم می‌توانید از روش تحلیلی (به دست آوردن روابط ریاضی) مسئله را حل کنید و اگر به پاسخ صحیح رسید نمره‌ی کامل را در هر دو حالت دریافت خواهید نمود.

۸- قابلیت اطمینان یک سیستمی توسط RBD به نحوی که در ادامه آمده است توصیف شده است مشخص کنید که TTF کل سیستم چه ارتباطی با TTF اجزاء دارد و چه تابعی از آن‌ها است. دقت کنید مشابه آنچه در سوال شماره‌ی ۶ داده شده است، TTF یک سیستم همواره تابعی از TTF اجزاء آن است.



موفق باشید

اجلالی