

به نام خدا



آزمایش شماره ۶

آزمایش معماری - دکتر سربازی آزاد

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی شیراز

نیمسال اول ۰۱-۰۰

گروه:

امیرحسین هادیان - ۹۷۱۰۲۶۰۹

محمدرضا مفیضی - ۹۸۱۰۶۰۵۹

علی حاتمی تاجیک - ۹۸۱۰۱۳۸۵



۱ هدف

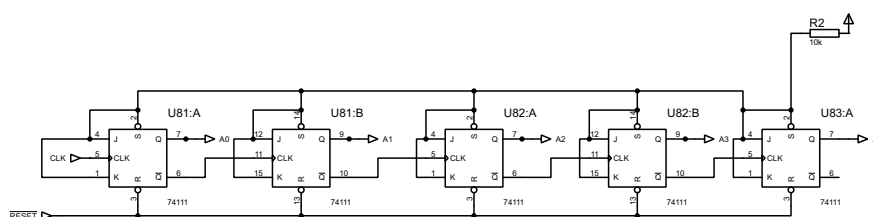
هدف از انجام این آزمایش متصل کردن بخش محاسبات طراحی شده در بخش قبل به یک حافظه برنامه‌پذیر است که برنامه محاسبه عدد دهم دنباله فیبوناچی است.

۲ طراحی

این بخش نیاز به طراحی خاصی نداشت چرا که از همان مدار آزمایش قبل استفاده شده است با این تفاوت که دیگر دستورات را به صورت دستی وارد نمی‌کنیم و از یک حافظه برنامه‌پذیر برای دادن برنامه به آن استفاده می‌کنیم. در این قسمت دو بخش اصلی داریم: اول، ساخت یک شمارنده ۵ بیتی و دوم، نوشتن برنامه مورد نظر به صورتی که بتوان آنرا در EPROM نوشت.

۱.۲ شمارنده ۵ بیتی

با استفاده از طراحی که در درس مدارهای منطقی برای شمارنده انجام داده‌ایم استفاده می‌کنیم و برای اختصار توضیح زیادی درباره آن نمی‌دهیم. این شمارنده یک شمارنده ۵ بیتی آسنکرون است که با استفاده از پنج فلیپ‌فلاپ JK ساخته شده است. شماتیک این شمارنده در شکل ۱ آمده است.



شکل ۱: شمارنده پنج بیتی

۲.۲ نوشتن برنامه

ابتدا با توجه به صورت دستور در آزمایش قبل کد را به زبان ماشین ترجمه می‌کنیم:

Address	Binary Code	Hex Code	Code
00000	00 1 00 000	20	SUB R0,R0
00001	00 0 01 101	0D	ADD R1,1
00010	00 0 00 001	01	ADD R0,R1
00011	00 0 01 001	09	ADD R1,R0
00100	00 0 00 001	01	ADD R0,R1
00101	00 0 01 001	09	ADD R1,R0
00110	00 0 00 001	01	ADD R0,R1
00111	00 0 01 001	09	ADD R1,R0
01000	00 0 00 001	01	ADD R0,R1
01001	00 0 01 001	09	ADD R1,R0

جدول ۱: کد مربوط به تولید عدد دهم فیبوناچی

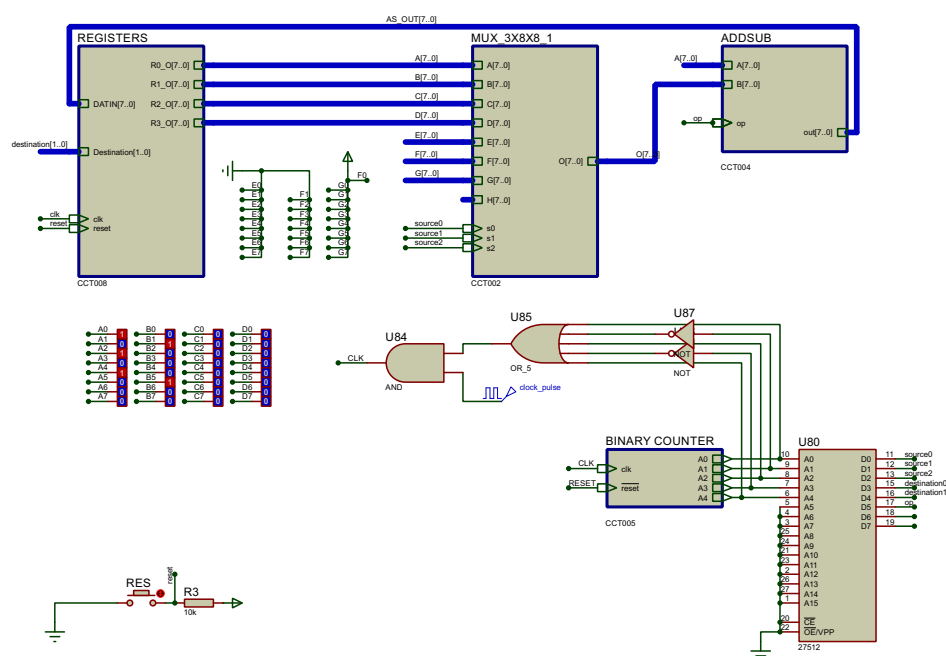
سپس با استفاده از نرم‌افزار HxD این کد هگزادسیمال ساخته شده را به صورت یک فایل هگز ۱۶ بیتی می‌نویسیم (این فایل یک نوع فایل متنی است که در ابتدای هر خط به فرمت خاصی عبارات نوشته می‌شوند و قابل استفاده برای شبیه سازی EPROM در پروتئوس خواهد بود. این فایل دارای محتویات زیر است که کد هگز ما درون آن مشخص است:



:0 A000000200D0109010901090109A1
:000000001FF

۳.۲ مدار نهایی

با اتصال شمارنده به حافظه برنامه‌پذیر و دادن برنامه به آن، خروجی‌های حافظه را به عنوان خطوط کنترلی واحد محاسبات که در آزمایش قبل ساخته بودیم استفاده می‌کنیم. شکل نشان‌دهنده مدار نهایی است. برای راحتی کار و اینکه نیازی نباشد تا برای هر دستور دکمه کلاک را فشار دهیم، از این ساز و کار استفاده می‌کنیم که با استفاده از یک گیت OR معکوس عدد ۱۰ (شماره برنامه‌ای که دیگر نباید اجرا شود) را با سیگنال کلاک AND می‌کنیم تا پس از آن کلاک به سیستم اعمال نشود و عملاً سیستم در حالت Idle قرار بگیرد. شکل ۲ نشان‌دهنده شمای کلی مدار است.



شکل ۲: مدار اصلی

۳ تست

نتیجه اجرای کد در شکل ۲ آمده است و واضح است که عدد ۳۴ درون رجیستر شماره ۱ وجود دارد. (رجیستر شماره ۰ حامل عدد ۲۱ است)