



پیاده‌سازی کامپیوتر پایه

قبل از طراحی ۱

برای پیاده‌سازی کامپیوتر پایه می‌توان از دو روش structural و behavioral استفاده کرد. در روش structural اجزای داخلی پردازنده را به صورت مجزا طراحی و پیاده‌سازی شده و در نهایت آن‌ها کنار هم قرار می‌گیرند. در روش behavioral، پردازنده توسط توصیف رفتار آن طراحی می‌شود. به مثال زیر دقت کنید:

مثال: دستور JP را به دو روش structural و behavioral بنویسید.

T0 : MAR \leftarrow PC ; MDR \leftarrow Mem[MAR]

T1 : IR \leftarrow MDR ; PC \leftarrow PC+1

T2 : PC \leftarrow IR

پاسخ: در روش structural ابتدا هر یک از اجزای PC, MAR, MDR, IR به صورت مجزا طراحی می‌شوند. سپس یک ماشین حالت فعال شدن هر یک از آن‌ها را در زمان‌های مختلف کنترل می‌کند. در نهایت تمامی این اجزا با باس داده مشترک به یکدیگر متصل می‌شوند و پردازنده را می‌سازند. ماشین حالتی که ترتیب و چگونگی مسیر داده را کنترل می‌کند، CU یا واحد کنترل نامیده می‌شود. طراحی این واحد کنترل به دو روش hardwired و micro programmed صورت می‌گیرد.

در طراحی behavioral تمامی اجزای پردازنده در یک فایل توصیف می‌شوند. رجیسترهای PC, MAR, MDR, IR به صورت سیگنال تعریف می‌شوند. ترتیب و چگونگی مسیر داده در خود پردازنده و در process توصیف می‌شود.

قبل از طراحی ۲

با به کارگیری rtl برای دستورات زیر، ترتیب انتقال رجیسترها را مشخص کنید.

Instruction	Operation
LD A, [ADDR]	ACC \leftarrow [ADDR]
ST A, [ADDR]	[ADDR] \leftarrow ACC
JP ADDR	JUMP TO ADDR
JPC ADDR	IF CF == 1: JUMP TO ADDR
AND A, [ADDR]	ACC \leftarrow ACC & [ADDR]
ADD A, [ADDR]	ACC \leftarrow ACC + [ADDR]
NOT A	ACC \leftarrow ~ACC
SHR A	ACC \leftarrow ACC >> 1

پیاده‌سازی کامپیوتر پایه

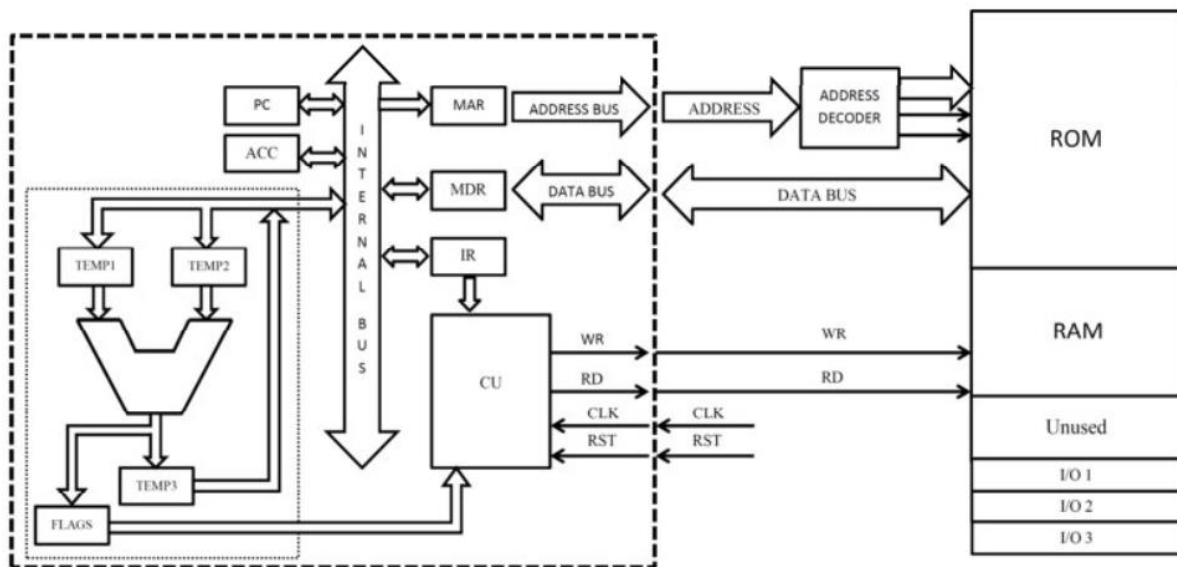
در این آزمایشگاه کامپیوتر پایه‌ای ۸ بیتی را طراحی خواهید کرد که از بخش‌های زیر تشکیل شده است:

- حافظه
- رمزگشای حافظه
- ورودی و خروجی
- پردازنده

پیاده‌سازی این کامپیوتر پایه در این آزمایش صورت می‌گیرد. در ابتدا پس از طراحی حافظه‌های بیرونی و گسترش حافظه در عرض، پهنای گذرگاه داده افزایش داده می‌شود. سپس این حافظه و تعدادی ورودی-خروجی در فضای آدرس‌دهی نگاشت می‌شوند. در ادامه با طراحی مدارهای کدگشا بخش پیرامونی پردازنده کامل می‌شود.

در ادامه بلوک‌های داخلی پردازنده طراحی شده و در انتها بلوک‌های طراحی شده در کنار یکدیگر قرار گرفته و کامپیوتر پایه را تشکیل می‌دهند.

شکل زیر بلوک دیاگرام پردازنده را نشان می‌دهد:



این پردازنده ۸ بیتی است و توانایی آدرس‌دهی ۳۲ بایت را دارد. تعداد خط‌های آدرس ۵ بیت و پهنای گذرگاه داده ۸ بیت است. هر دستور در این پردازنده یک بایت طول دارد و شامل دو بخش زیر است:

- Opcode: سه بیت پر ارزش دستور

- Operand: پنج بیت کم ارزش دستور

پردازنده شامل بلوک‌های داخلی زیر است که این بلوک‌ها توسط باس داخلی ۸ بیتی با هم ارتباط دارند:

- رجیستر دستور IR: رجیستری که داده دستور را از باس داخلی بارگذاری کرده و opcode را به cu و operand را به باس داخلی تحویل می‌دهد.
- رجیستر داده MDR: رجیستر دو طرفه‌ای که داده را میان باس داخلی و باس داده خارجی قرار می‌دهد.
- رجیستر آدرس MAR: رجیستری که داده آدرس را از باس داخلی بارگذاری کرده و به طور مستقیم بر باس خارجی قرار می‌دهد.
- شمارنده برنامه PC: رجیستری است که آدرس دستور بعدی در آن قرار می‌گیرد.
- انبار ACC: رجیستری که می‌تواند داده را از باس داخلی بگیرد یا بر آن قرار دهد.
- واحد کنترل CU: بخش‌های گوناگون پردازنده را با هم هماهنگ می‌کند.
- ALU: محاسبات عددی و منطقی در این بخش صورت می‌گیرد.

۱- طراحی حافظه‌های مورد نظر

یک حافظه ROM شانزده بیتی طراحی کنید که شامل سیگنال rd، cs، خروجی یک بیتی و باس آدرس ۴ بیتی است.

یک حافظه RAM هشت بیتی طراحی کنید که شامل سیگنال rd، wr، cs، خروجی یک بیتی و باس آدرس ۳ بیتی است.

۲- طراحی مدارهای رمزگشایی حافظه و ورودی-خروجی

یک مدار رمزگشای حافظه طراحی کنید که خروجی‌های زیر را به ازای بازه‌های موردنظر بدهد:

Address Range	Signal
00h – 0fh	Chip_Select_ROM
10h – 17h	Chip_Select_RAM
1dh	Chip_Select_IO0
1eh	Chip_Select_IO1
1fh	Chip_Select_IO2

یک درگاه خروجی، یک درگاه ورودی و یک درگاه ورودی-خروجی طراحی کنید.

درگاه خروجی دارای سیگنال یک بایتی ورودی باس داده، سیگنال یک بایتی خروجی و دو سیگنال تک بیتی cs و wr است. هنگامی که cs و wr فعال باشد، مقدار موجود در باس داده بر روی پورت خروجی قرار می گیرد.

درگاه ورودی دارای سیگنال یک بایتی ورودی، سیگنال یک بایتی خروجی باس داده و دو سیگنال تک بیتی cs و rd است. هنگامی که cs و rd فعال باشد، مقدار موجود در پورت ورودی بر روی باس داده قرار می گیرد.

درگاه ورودی-خروجی دارای یک سیگنال یک بایتی ورودی-خروجی، یک سیگنال یک بایتی ورودی-خروجی باس داده، سیگنال یک بایتی خروجی و سه سیگنال تک بیتی wr، cs و rd است. این پورت دو طرفه می تواند هم ورودی و هم خروجی باشد.

۳- طراحی رجیسترهای داخلی

رجیسترهای داخلی زیر را طراحی کنید:

ACC شامل:

- سیگنال دوطرفه ۸ بیتی D
- سیگنال ورودی تک بیتی LE
- سیگنال ورودی تک بیتی OE

MAR شامل:

- سیگنال ورودی ۸ بیتی D
- سیگنال خروجی ۵ بیتی A
- سیگنال ورودی تک بیتی LE

MDR شامل:

- سیگنال دوطرفه ۸ بیتی DIN
- سیگنال دوطرفه ۸ بیتی DOUT
- دو سیگنال ورودی تک بیتی LE_IN و LE_OUT
- دو سیگنال ورودی تک بیتی OE_IN و OE_OUT

IR شامل:

- سیگنال دوطرفه ۸ بیتی D
- سیگنال خروجی ۳ بیتی I
- سیگنال ورودی تک بیتی LE
- سیگنال ورودی تک بیتی OE

PC شامل:

- سیگنال دوطرفه ۸ بیتی D
- سیگنال ورودی تک بیتی LE
- سیگنال ورودی تک بیتی OE
- سیگنال ورودی تک بیتی CLR
- سیگنال ورودی تک بیتی CLK
- سیگنال ورودی تک بیتی INC

۴- طراحی ALU

واحد محاسبه و منطق را طوری طراحی کنید که دارای دو رجیستر ورودی ۸ بیتی temp1 و temp2 و یک رجیستر ۸ بیتی temp3 باشد. همچنین شامل:

- سیگنال دوطرفه ۸ بیتی D
- سیگنال خروجی تک بیتی CF
- سیگنال ورودی دو بیتی C
- سه سیگنال ورودی تک بیتی LE_temp1، LE_temp1 و LE_temp1
- سیگنال ورودی تک بیتی OE

واحد ALU با کدهای کنترلی C عملیات زیر را انجام می‌دهد:

C[1:0]	Operation
00	AND (TEMP3 ← TEMP1 & TEMP2)
01	ADD (TEMP3 ← TEMP1 + TEMP2 , CF ← carry)
10	NOT (TEMP3 ← ! TEMP1)
11	SHR (TEMP3 ← {0, TEMP1[7:1]})

۵- طراحی CU

واحد کنترلی با روش hardwired طراحی کنید که شامل دستورات زیر باشد:

OP-Code	Instruction	Operation
000	LD A, [ADDR]	$ACC \leftarrow [ADDR]$
001	ST A, [ADDR]	$[ADDR] \leftarrow ACC$
010	JP ADDR	JUMP TO ADDR
011	JPC ADDR	IF CF == 1: JUMP TO ADDR
100	AND A, [ADDR]	$ACC \leftarrow ACC \& [ADDR]$
101	ADD A, [ADDR]	$ACC \leftarrow ACC + [ADDR]$
110	NOT A	$ACC \leftarrow \sim ACC$
111	SHR A	$ACC \leftarrow ACC \gg 1$

۶- تکمیل پردازنده

طراحی‌های بخش ۳ تا ۵ را بهم متصل کنید تا پردازنده کامل شود.

۷- تکمیل کامپیوتر پایه

برای اتمام کامپیوتر پایه، پردازنده تکمیل شده در بخش ۶ را به بخش ۱ و ۲ متصل کنید.

با کمک جدول ارائه شده در بخش ۵، برنامه‌ای بنویسید که دو مقدار از ورودی خوانده و میانگین آن دو را در درگاه خروجی بنویسد. کد نهایی را در ROM قرار دهید.

\$- بخش امتیازی

کامپیوتر پایه‌ای که به صورت structural طراحی کردید را بهینه کرده و به صورت رفتاری توصیف کنید.

** در هر بخش برای اطمینان از صحت اجرای کدهای نوشته شده می‌توانید با نوشتن فایل آزمون مناسب عملکرد آن را ارزیابی کنید.

موفق باشید

اردی‌بهشت ۹۷

قاسمی