

گزارش کار ۱۲

سیهر مقیسه

روح الله احمدزاده

۹۸۳۱۱۰۳

۹۸۳۱۰۰۱

■ موجودیت microprocessor دارای یک سری input و output است :

reset

برای reset کردن دستگاه

Clk

برای کلاک دستگاه

Data

خطوط داده

Address

خطوط ادرس

Memr-memw

برای خواندن از حافظه و نوشتن به آن

Input port

پورت های ورودی

Outport

پورت های خروجی

Intr_in

برای ورودی وقفه که نشان دهنده این است یک دستگاه جانبی نیاز به cpu دارد

Intr_out

برای خروجی وقفه-که اگر cpu جواب دهد 1 در غیر این صورت 0 است

■ در بخش معماری microprocessor یک سری سیگنال برای ثبات ها تعریف شده است که هر کدام کار مشخصی دارند:

Dr

ثبات نگه داری داده که به اندازه یک word است

Ar

ثبات نگه داری آدرس که به اندازه تعداد ردیف های حافظه است

Ac

ثبات انباشتگر که نتیجه alu رو نگه میدارد

Ir

ثبات دستور العمل که شامل آدرس ثبات-کد دستورالعمل-بیت غیر مستقیم و یا مستقیم میشود

Pc

ثبات شمارنده برنامه که آدرس دستور العمل بعدی را نگه میدارد

Tr

ثبات موقت که برای نگه داری بعضی داده ها برای بعضی دستورالعمل ها نیاز است

Inpr

ثباتی که اطلاعات را از کامپیوتر گرفته و به cpu می‌رساند

Outr

ثباتی که اطلاعات را از cpu گرفته و به کامپیوتر می‌رساند

■ در قسمت بعدی معماری سیگنال های داخلی cpu ساخته شده اند :

D

خروجی دیکود شده دستورالعمل

Sc

شمارنده که برای ایجاد سیگنال های زمانی و همچنین شمارش تعداد بار هایی که یک رویداد اتفاق می افتد به کار می آید.

T

سیگنال های زمانی که از کلاک کامپیوتر به وجود می آیند که میتواند به سايكل های کندتری تبدیل شود

I

بیت مستقیم یا غیر مستقیم

e

ثبات انباشت گر جانبی

S

شروع و پایان فلیپ فلاپ

En_id

بیتی که برای فعال شدن دیکودر استفاده میشود

Clr_sc

پاکسازی شمارنده

Fgi

فلگ ورودی که هر وقت اطلاعات جدیدی در ثبات inpr باشد ۱ میشود و هر وقت توسط کامپیوتر خوانده و قبول شد 0 میشود

Fgo

هر وقت که آخرین کرکتر خوانده شد توسط i/o 1 میشود

len

سیگنالی که نشان میدهد وقفه ایجاد شده و فلیپ فلاپ فعال شدن وقفه نام دارد

R

فلیپ فلاپ وقفه

■ Instruction decoder

در این بخش

یک دیکودر ۳ به ۸ وجود دارد که ۳ بیت opcode را گرفته و تبدیل به ۸ بیت میکند تا یکی از ۸ دستور احتمالی موجود توسط کامپیوتر انجام شود و توسط بیت en_id مدیریت میشود (اگر یک بود فعال میشود)

■ 4-bit sequence counter

هر زمان که فلیپ فلاپ غیر فعال شد یک عدد به شمارنده اضافه میشود لازم به ذکر است اگر بیت clr_sc یک بود شمارنده ریست میشود

■ 4 to 16 decoder for timing signals

از شمارنده ۴ بیتی برای ایجاد سیگنال های زمانی استفاده میشود که میتوان از سایکل های کلاک استفاده بیشتری کرد و به اصطلاح کند تر کرد

■ Control unit

Reset microprocessor

در این بخش اگر بیت reset یک بود تمام سیگنال های ورودی microprocessor پاکسازی میشوند

Load ar with contents of pc

در این بخش اگر در زمان t0 بودیم محتوای pc را در ar ریخته و از ادرس ar که به یک خانه حافظه اشاره دارد مقدار داده را میخوانیم

Fetch instruction and increment pc

در اینجا pc را یک عدد زیاد کرده تا به دستور العمل بعدی اشاره کند و سپس دیتا خوانده شده را به دستور العمل میدهیم

Decode opcode

در اینجا بیت خواندن از حافظه را به پایان رسانده و همچنین به flag ها دستور می‌دهیم که حواسشان به وقفه های ورودی و خروجی باشند

و سپس بیت دیکودر را فعال کرده که سه بیت opcode توسط دیکودر کدگذاری شود تا یکی از دستور هارا استفاده کنیم

همچنین با ارزش ترین بیت را برای متوجه شدن مستقیم یا غیر مستقیم بودن دستور به سیگنال I می‌دهیم

Store return address in tr

در اینجا ما باید عملیات انجام شده را در ادرس برگشت که از حافظه است، در ثبات موقت ثبت می‌کنیم

Store return address in location 0

در اینجا آدرس ثبت شده در ثبات موقت را به دیتا می‌دهیم که نتیجه دستور العمل انجام شده را در همان ادرس از حافظه ثبت کنیم و سپس در پایان دستور های داده شده مقدار pc را صفر می‌کنیم

Increment pc reset ien and r

در اینجا اگر همچنان دستورات مانده باشد

Pc را یک عدد زیاد کرده تا دستور بعدی را پیدا کنیم و سپس interrupt enable را صفر می‌کنیم و همچنین ثبات i/o را پاکسازی می‌کنیم

Execute i/o instruction

در صورت دریافت سیگنال interrupt از سمت یکی از I/O ها وارد این قسمت می‌شویم و به دنبال دستگاهی می‌گردیم که سیگنال را فرستاده. سپس ورودی را دریافت یا خروجی را تحویل می‌دهیم.

INP (input character)

ورودی گرفتن یک کارکتر

OUT (output character)

خروجی دادن یک کارکتر

SKI (skip on input flag)

در نظر نگرفتن ورودی

SKO (skip on output flag)

در نظر نگرفتن خروجی

ION (Interrupt enable on)

رفتن به حالت پذیرش سیگنال interrupt

IOF (interrupt enable off)

رفتن به حالت عدم پذیرش سیگنال interrupt

Execute register reference instruction

اجرای دستورات ثباتی

CLA (clear ac)

صفر کردن مقدار ثبات ac

CLE (clear e)

صفر کردن پرچم E

CMA (complement ac)

نقیض کردن مقدار ثبات ac

CME (complement e)

نقیض کردن پرچم E

CIR (ciculate right)

یک شیفت دورانی به سمت راست روی ثبات ac

CIL (ciculate left)

یک شیفت دورانی به سمت چپ روی ثبات ac

INC (increment ac)

افزایش مقدار ثبات ac به اندازه یکی

SPA (skip if positive)

رد کردن دستور در صورت مثبت بودن ac

SNA (skip if negative)

رد کردن دستور در صورت منفی بودن ac

SZA (skip if zero)

رد کردن دستور در صورت صفر بودن ac

SZE (skip if e is zero)

رد کردن دستور در صورت صفر بودن E

HLT (halt)

مکث

Fetch address for indirect addressing

گرفتن آدرس operand برای حالت آدرس دهی غیر مستقیم

Do nothing for direct addressing mode

هیچ کاری نکردن برای حالت آدرس دهی مستقیم

AND (and to ac)

And گرفتن با ac

ADD (add to ac)

اضافه کردن به ac

LDA (load to ac)

خواندن از حافظه و ریختن در ac

STA (store ac)

ذخیره کردن مقدار ثبات ac در حافظه

BUN (branch unconditionally)

پرش بدون شرط

BSA (branch and save return address)

پرش و ذخیره کردن آدرس برگشت

ISZ (increment and skip if zero)

افزایش و رد کردن در صورت صفر بودن

Memory read for AND, ADD, LDA & ISZ instructions

And گرفتن با ac