

به نام خدا

تمرین ۱۰ درس ریز پردازنده

امیر محمد پیر حسین لو

۹۵۳۱۰۱۴

1- لچ دارای حافظه است در صورتی که بافر حافظه ندارد. لچ می تواند مقدار خروجی خود را با توجه به ورودی هایی که به آن می دهیم تا هر مدت زمانی حفظ کند اما بافر چنین قابلیتی ندارد. نمی توان جای لچ و بافر را عوض کرد.

2- لچ:

tSHSL: زمان ۱ بودن strobe

tIVSL: فاصله بین زمان گذاشتن داده تا لبه پایین رونده سیگنال strobe

tSLIX: فاصله بین زمان پایین رفتن پالس strobe تا انتهای زمان معتبر بودن داده بافر:

tPLH: تاخیر انتشار برای تغییر وضعیت خروجی از ۰ به ۱ با فرض اینکه پایه های Enable از قبل فعال بوده اند.

tPHL: تاخیر انتشار برای تغییر وضعیت خروجی از ۱ به ۰ با فرض اینکه پایه های Enable از قبل فعال بوده اند.

tPZL: تاخیر انتشار برای رفتن خروجی از حالت شناور به وضعیت ۰

tPZH: تاخیر انتشار برای رفتن خروجی از حالت شناور به وضعیت ۱ دیکودر:

tPHL: تاخیر انتشار از زمان گذاشتن آدرس تا تغییر وضعیت خروجی از ۱ به ۰ با فرض این که پایه های Enable از قبل فعال بوده اند.

tPLH: تاخیر انتشار از زمان گذاشتن آدرس تا تغییر وضعیت خروجی از ۰ به ۱ با فرض این که پایه های Enable از قبل فعال بوده اند.

برای انکودر هم مانند دیکودر است با این تفاوت که ورودی می دهیم و آدرس می گیریم.

3- در روش سرکشی به طور مداوم در حال چک کردن یک پرچم (متغیر) هستیم تا ببینیم درخواستی برای انجام کاری وجود دارد یا خیر. در این حالت کار دیگری نمی توان انجام داد و بازدهی سیستم به شدت پایین است زیرا تنها در صورت وجود درخواست، کار انجام می شود و در بقیه حالات در حال از دست دادن انرژی برای چک کردن تنها یک پرچم هستیم. در روش مبتنی بر وقفه، تنها زمانی که وسیله جانبی درخواستی برای اجرای کاری داشته باشد، ریزپردازنده وضعیت فعلی خود را ذخیره میکند و سراغ اجرای کار دستگاه مربوطه می رود.

در سایر اوقات که درخواست وجود ندارد، ریزپردازنده کارهای مفید دیگری انجام می دهد و این یعنی این که در هر لحظه در حال اجرای کار مفید هستیم که باعث می شود کارایی بیشینه شود.

4- دو راه وجود دارد:

a. استفاده از Encoder که با توجه به جدول حالات آن می توان برای اجرای وقفه ها اولویت در نظر گرفت.

b. پایه خروجی نشان دهنده درخواست از دستگاه درخواست دهنده وقفه که به گیت منطقی (AND, OR, ...) وصل شده است را به یکی از پایه های میکرو وصل می کنیم. حال اگر وقعه آمد، به ترتیب آن پایه هایی از میکرو که به آن پایه های دستگاه های درخواست دهنده IO متصل شده است را به ترتیب چک می کنیم و درخواست ها را بررسی می کنیم.

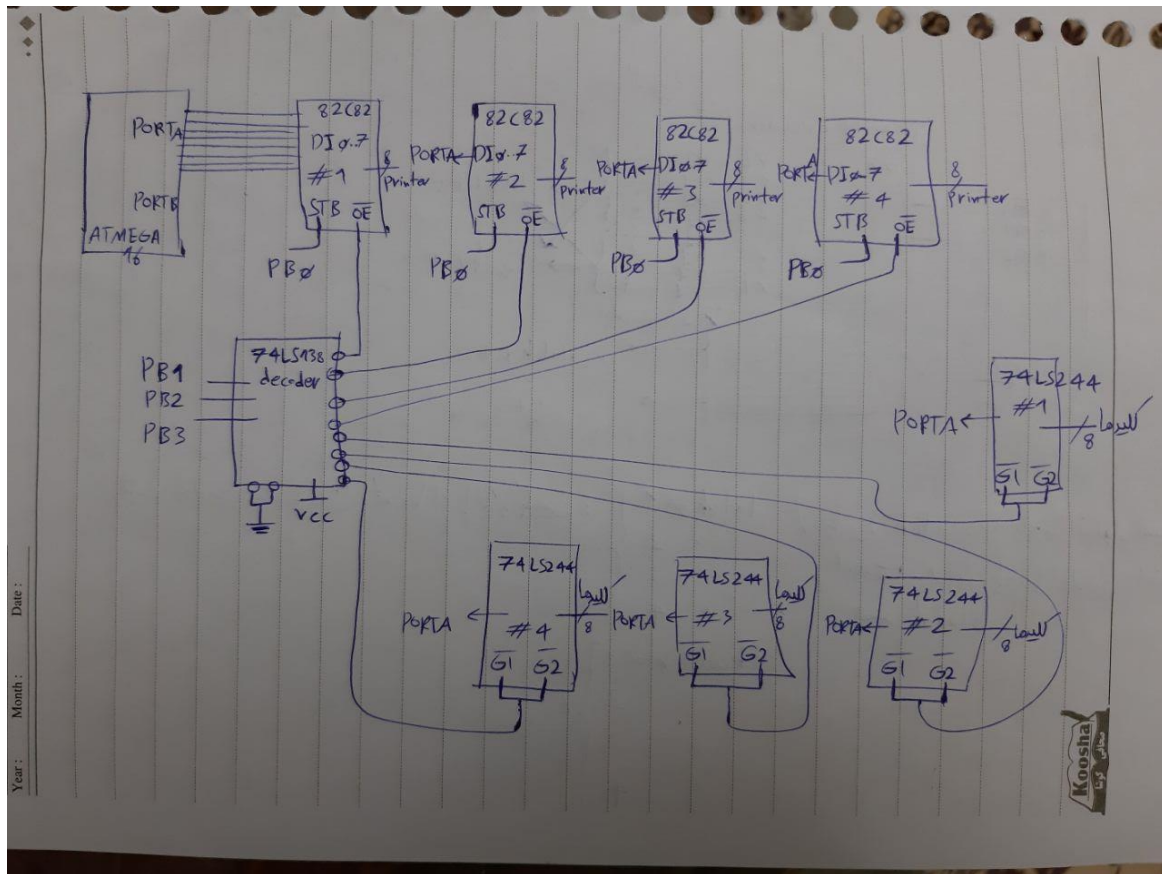
5- بله، می توان ترتیب چک کردن پرچم ها (متغیرها) را به ترتیب اولویت دستگاه قرار داد. مانند مثال زیر:

```
; Polling of 8 devices
; Busy Line of all 8 devices are connected to PB

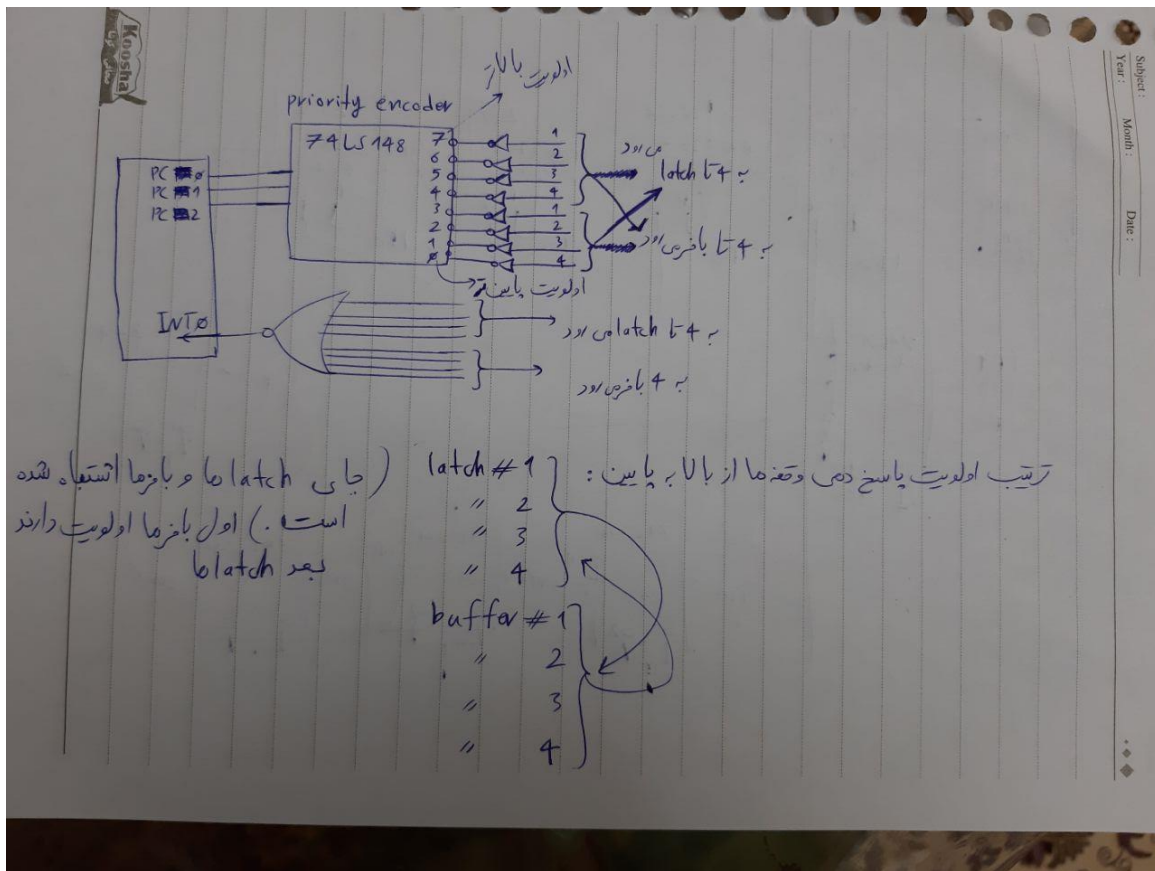
LOOP1:      SBIS      PINB, 0      ; Skip next inst. if FD is not Ready      [1/3]
             RCALL     FD          ;                               [3]
             SBIS      PINB, 1      ; Skip next inst. if LP is not Ready      [1/3]
             RCALL     LP          ;                               [3]
             SBIS      PINB, 2      ; Skip next inst. if DWP is not Ready      [1/3]
             RCALL     DWP         ;                               [3]
             SBIS      PINB, 3      ; Skip next inst. if PLOT is not Ready      [1/3]
             RCALL     PLOT        ;                               [3]
             SBIS      PINB, 4      ; Skip next inst. if ADC is not Ready      [1/3]
             RCALL     ADC         ;                               [3]
             SBIS      PINB, 5      ; Skip next inst. if DAC is not Ready      [1/3]
             RCALL     DAC         ;                               [3]
             SBIS      PINB, 6      ; Skip next inst. if MOD is not Ready      [1/3]
             RCALL     MOD         ;                               [3]
             SBIS      PINB, 7      ; Skip next inst. if TERM is not Ready      [1/3]
             RCALL     TERM        ;                               [3]
             JMP       LOOP1       ;                               [3]
```

6- PC و status register باید ذخیره شود و سپس بازیابی شود. در ATMEGA16، PC به صورت خودکار ذخیره می شود اما status register را باید خودمان ذخیره کنیم. ذخیره سازی در حافظه پشته با توجه به SP (اشاره گر پشته) انجام می شود.

7- .



- a. محل کلید ها و پرینترها در شکل قسمت الف مشخص شده است.
- b. کد در فایل 7-b.asm موجود است.



c.

d. کد در فایل 7-d.asm موجود است.

تراشه 74148

INPUTS										OUTPUTS				
EI	0	1	2	3	4	5	6	7		A2	A1	A0	GS	EO
H	X	X	X	X	X	X	X	X		H	H	H	H	H
L	H	H	H	H	H	H	H	H		H	H	H	H	L
L	X	X	X	X	X	X	X	L		L	L	L	L	H
L	X	X	X	X	X	X	L	H		L	L	H	L	H
L	X	X	X	X	L	H	H	H		L	H	L	L	H
L	X	X	X	L	H	H	H	H		L	H	H	L	H
L	X	X	L	H	H	H	H	H		H	L	L	L	H
L	X	L	H	H	H	H	H	H		H	L	H	L	H
L	L	H	H	H	H	H	H	H		H	H	H	L	H

شکل ۲۴- بلوک دیاگرام عملیاتی تراشه 74LS148