



زمان پاسخ: ۱۲۰ دقیقه

سئوالات پایان ترم درس ریزپردازنده و زبان اسمبلی - نیمسال دوم سال تحصیلی ۱۴۰۱ - ۱۴۰۲

نقش هر کدام از رجیسترهای زیر را به اختصار شرح دهید:

- BSR a ✓
STATUS b ✓
FSR c ✓
TABLAT d ✓

۲۰

۱۴۰۰۵۳۸ ۱۴۰۴۸

۲ به زبان اسمبلی PIC18 برنامه ای بنویسید که محتوای خانه 0x10 حافظه را بخواند و parity آن را محاسبه کند (تعداد بیت های 1 آن را بشمارد) اگر زوج باشد بیت اول از PORTC را 1 کند و در غیر این صورت 0 کند

۲۰

۳ برنامه ای بنویسید که n امین جمله سری فیبوناچی را محاسبه کند. سری فیبوناچی به دنباله ای از اعداد می گویند که به صورت زیر تعریف می شود

۲۰

$$F(n) := \begin{cases} 1 & \text{if } n = 1; \\ F(n-1) + F(n-2) & \text{if } n > 1. \end{cases} \quad \begin{matrix} F(1) = 1 \\ F(2) = 1 \end{matrix}$$

غیر از دو عدد اول، اعداد بعدی از جمع دو عدد قبلی خود به دست می آیند

0, 1, 1, 2, 3, 5, 8, 13,

۴ فرض کنید دو عدد ۸ بیتی بدون علامت در خانه های حافظه داده به شماره های 0x10, 0x11 قرار دارند. برنامه ای بنویسید که به روش تفریق های متوالی عدد اول را به دومی تقسیم کرده و خارج قسمت را در خانه 0x12 و باقیمانده را در خانه 0x13 قرار دهد

۲۰

$$\frac{\text{اول}}{\text{دوم}} = \square \square$$

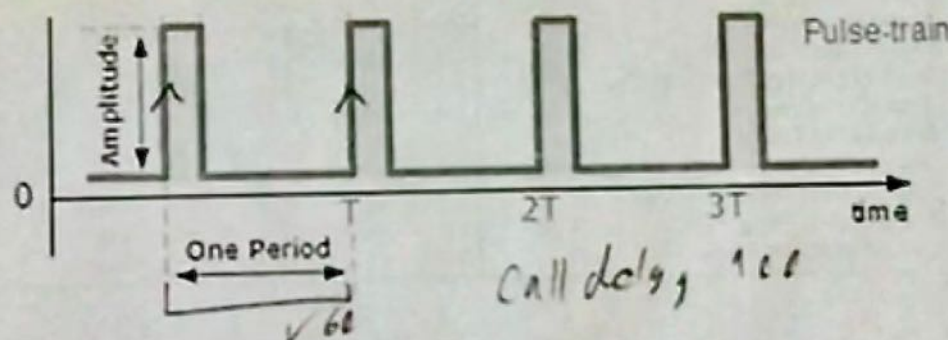
INT0/RBO

۵ فرض کنید یک پالس مربعی به پایه RBO میکروکنترلر اعمال می شود. برنامه ای بنویسید که پریود این پالس را با دقت میلی ثانیه حساب کرده و روی PORTC قرار دهد.

4ms ± 2

BSF INTCON:INTEDG0 ✓

۲۰



CPFSGT

تمامی برنامه ها بایستی به زبان اسمبلی نوشته شود. رجیسترها و متغیرهای مورد نیاز را تعریف کرده و پیگردانی های سخت افزاری لازم را انجام دهید.

موفق باشید - معلمی خیاوی

① **BSR** یک رجیستر 4 بیتی بوده که با کمک آن می توانیم تعیین کنیم که داده (خانه حافظه) در نظر در کدام یک از 16 بانک حافظه Memory RAM، 4KB، قرار دارد. رجیستر این رجیستر استفاده می شود که بیت 0، دستور العمل 1 شده باشد آن وقت از بانک BSR برای بانک access استفاده می کنیم.

Bank Select Register = BSR <3:0>

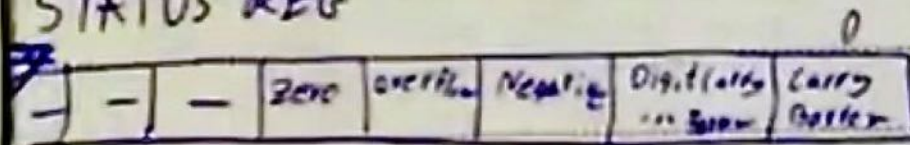
MOVLB 0x2

MOVWF 0x10, BANKED (1) $\Rightarrow W \rightarrow 0x210$

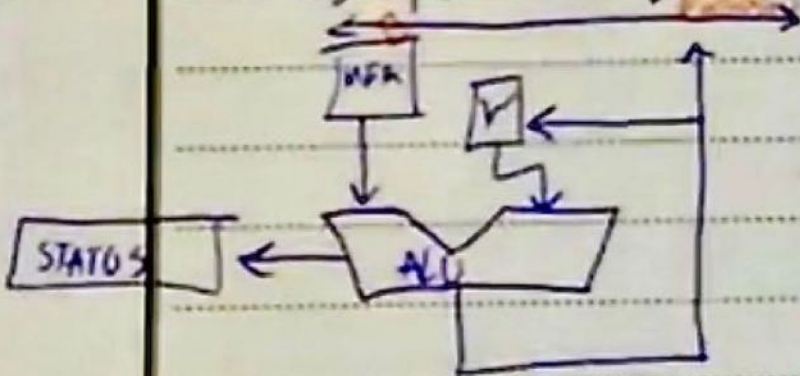


② رجیستر وضعیت یک رجیستر 8 بیتی بوده که تنها 5 بیت آن استفاده می شود که به ترتیب هر بیت نشان دهنده وضعیت خاص خود است:

STATUS REG



هر یک از بیت های این رجیستر زمانی تغییر پیدا می کند که عملیات با ALU انجام شود:



ما می توانیم با استفاده از دستورات بیت های خاص را حرکت دهیم و حتی تغییر دهیم:

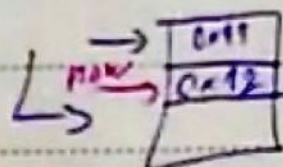
BTFSS STATUS, C

③ **FSR** File Select Register، با استفاده از رجیستر فدرق می توانیم به — خانه ای که FSR برای آن استفاده می کند دست پیدا کنیم. FSR برای دسترسی دینا مکتب به نام های حافظه و همچنین پیکره های آرایه ها کاربرد فراوانی دارند و با آن می توانیم به طور غیر مستقیم به خانه های حافظه دسترسی پیدا کنیم.

LF SR 0, 0x11

MOVWF INDF0

INCF FSR0



با استفاده از این رجیستر می‌توانیم عناصر جدول Look up Table را بخوانیم و در حقیقت این رجیستر همان خانه کفزی است که با استفاده از TBLRD⁺ می‌توانیم به آن دسترسی داشته باشیم و محتوای آن را بخوانیم و تغییر دهیم. به این رجیستر جدول جدول هم گفته می‌شود. لازم به ذکر است پس از این دستور TBLRD⁺، خانه بعدی در TACHT می‌تواند به جدول دسترسی داشته باشد.

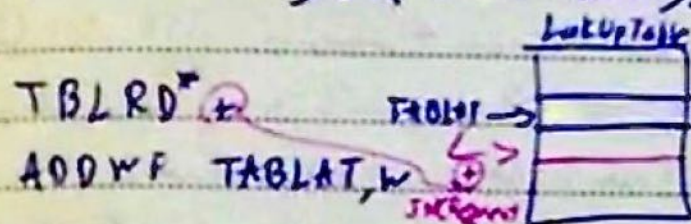


Table Reads

Table Latch

#include <pic14f452.h>

LIST _P=14F452

CONFIG WACHDOG=OFF

CONFIG OSC=XT → RA01X DEC

CNT EQU 0x2

→ Assembler directives

CNT_2 EQU 0x5 → 0x05

CIRF TRISC

CIRF CNT_2

MOVLW 0

MOVWF CNT

BTFSC 0x10, 0

INCF CNT_2, F

RRNCF 0x10, F

DECFSZ CNT, F

BRA loop

BTFSS CNT_2, 0

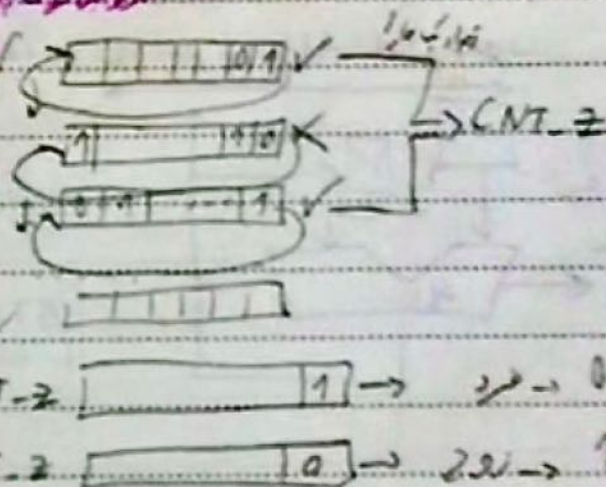
BSF PORTC, 0

BCF PORTC, 1

HH:

GOTO HH

END



نمودار رجیستر CNT و CNT_2
محتوای رجیستر CNT در ابتدا 0 است.
در هر بار که دستور INCF اجرا شود، محتوای رجیستر CNT یک واحد افزایش می‌یابد.
دستور BTFSC 0x10, 0 بررسی می‌کند که بیت 0 رجیستر 0x10 (PORTC) صفر است یا نه.
اگر بیت 0 صفر باشد، دستور INCF اجرا می‌شود و محتوای رجیستر CNT یک واحد افزایش می‌یابد.
دستور RRNCF 0x10, F رجیستر 0x10 (PORTC) را یک بیت به راست می‌رخساند.
دستور DECFSZ CNT, F رجیستر CNT را یک واحد کم می‌کند و اگر به صفر نرسد، به دستور BTFSC می‌رود.
دستور BTFSS CNT_2, 0 بررسی می‌کند که بیت 0 رجیستر CNT_2 صفر است یا نه.
اگر بیت 0 صفر باشد، دستور BSF PORTC, 0 بیت 0 رجیستر PORTC را یک به یک می‌گذارد.
اگر بیت 0 یک باشد، دستور BCF PORTC, 1 بیت 1 رجیستر PORTC را یک به یک می‌گذارد.
دستور GOTO HH به خط HH می‌پرش.
دستور END پایان برنامه است.

if $CNT == 2$ Result = 1 | if $CNT == 1 \rightarrow Result = 0$
if $CNT > 2 \rightarrow CNT = CNT - 1$
✓ $CNT = 5$ ✓

③ برای یافتن 7 امین عدد سری فیبوناچی:

F.1. EGV 0.4

Ed EGV Dy 3

RESULT EQU 0x7

Next: $D's$
SUBWT CNT, F = INT ← CNT - 2

MOVLW 0'0'

max WF $E_Q \rightarrow$ اولین عدد است

Nov 18 1911

max F $F_0 \rightarrow$ max cost

W. I. M. F. F. Q. W.

App VF E, W

MOV FF, F1, F0

MOYNE F1

DECLF52 CNT

TRA Loop

MOVFF F, RESULT

FINISH:

GOTO FINISH

END

MSKTLW D'9'

CPFS@CNT

BR A COM

↳ **VEER F RESULT**

GO TO FINISH
LAN: MOVLO'D

CPFS ~~LR~~ LNT

BRA NEAT

→ MOV 2 0'1

MOVWF RA
RSTC EINT

CM - (دوسرا حصہ)

"

Res

[illegible]

و این را به صورت RE می بینیم

(4)

اول: $0x11 \rightarrow 0x12$
دوم: $0x10 \rightarrow 0x11$

$4/3 = 1 \times 3 + 1$
 $2/3 = 0 \times 3 + 2$
باقی ماند 2
CPFSLT
 $F < 0$

CNT EQU 0x6 \Rightarrow خارج قسمت ✓

MOV F 0x10

$7/3 = 2 \times 3 + 1$
باقی ماند 1
خارج قسمت 2
باقی ماند 1

loop: MOV F 0x10, W

CPFSLT 0x11

GRA CAN

GRA FINISH

$0x11 > 0x10$

$0x11 < 0x10$

LDV:

برای تفهیم

MOV F 0x10, F

SWWF 0x11, F $(0x11 - 0x10) \rightarrow 0x11$

INCF CNT, F $CNT++$

DRA loop

حالتی که
وقت را بگذرد

FINISH:

MOV F CNT, 0x12 \rightarrow خارج قسمت

MOV F 0x11, 0x13 \rightarrow باقی ماند

نوع برنامه: ...
 ORG 0x00
 GOTO 35H

دانشگاه ...
 دانشکده ...

شماره دانشجویی: 140053611048
 سال تحصیلی: ...
 تاریخ برگزاری امتحان: ...

MAIN: CNT EQU 0x04

CLRF TRISC
 BCF RCN, IPEN
 BSF INTCON, INTOIE
 BSF INTCON2, INTERGO
 BSF INTCON, GIE
 CLRF CNT, F
 CLRF PORTC
 Loop: CALL DELAY
 INCF CNT, F
 BRF Loop

35H

BTFS INTCON, INTOIE
 RETFIE
 MOV EF CNT, PORTC
 BCF INTCON, INTOIE
 CLRF CNT
 RETFIE

140053611048

END

$f_{clk} = 4MHz \rightarrow T_{cycle} = 1\mu s$

100ms Delay = 1000 میکروثانیه

Delay: { NOP }
 1000

RB0 \Rightarrow INTO
 External Interrupt Device

1000 = 10 * 100

DELAY:

MOVLW D'1
 MOVWF CNT-1

Loop1: MOVLW D'1
 MOVWF CNT-2

Loop2: DECFSZ CNT-2
 BRA Loop2

DECFSZ CNT-1 ... 2
 BRA Loop1 ... 2

RETURN

1ms = 1000 میکروثانیه

1ms = 1000 * 10⁻⁶ (ثانیه)