

Resolución de tarea de Electrónica Digital II

Valenzuela Gabriel

Universidad Nacional de Córdoba - FCEFYN

23 de marzo de 2019

1. Ejercicio 2.1

El PIC 16F887 posee una arquitectura de PIC de escala media. Esta, posee 35 instrucciones, cada una de 14bits de largo. En cuanto a la memoria, podemos decir que:

- La memoria del programa está organizada en páginas de 2048 words.
- La memoria de datos está formada por registros de 8bits, organizada en bancos de 120 registros.
- Esta arquitectura puede llegar a tener hasta 4 bancos.

En general, estos microcontroladores, poseen alguna memoria de datos EEPROM y una stack de 8 niveles para almacenar la dirección de la memoria de programa.

Estos PIC poseen un sistema de interrupciones fijas para interrupciones internas (De sus sistemas internos) y una para interrupción externa. Cada bloque de I/O puede generar una solicitud de interrupción a la CPU, pueden recibir estas solicitudes desde dispositivos externos. Los dispositivos de I/O poseen varios puertos paralelos (A,B,C,etc); 3 timers, 2 módulos de adquisición, comparación, y PWM; varios puertos seriales para comunicación síncrona como asíncrona, un ADC de 10bit asociado a un multiplexor analógico. La figura siguiente muestra la arquitectura interna.

■ **Banco 3:** 0x190 – 0x1EF

Tomando en cuenta que cada GPRs cuenta con 96 bits, el programador dispone de 48Bytes. En cuanto a los registros de acceso se destacan el *RP1*, *RP0* para acceso directo y el *IRP* para acceso indirecto.

RP1	RP0
0	0
0	1
1	0
1	1

Donde se accede al Banco 0, 1, 2 y 3 respectivamente mientras que para el IRP en 0 se accede al Banco 0 o 1 y en 1 al Banco 2 o 3.

File	Address	File	Address	File	Address	File	Address
Indirect addr. ⁽¹⁾	00h	Indirect addr. ⁽¹⁾	80h	Indirect addr. ⁽¹⁾	100h	Indirect addr. ⁽¹⁾	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h	WDTCON	105h	SRCON	185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
PORTC	07h	TRISC	87h	CM1CON0	107h	BAUDCTL	187h
PORTD ⁽²⁾	08h	TRISD ⁽²⁾	88h	CM2CON0	108h	ANSEL	188h
PORTE	09h	TRISE	89h	CM2CON1	109h	ANSELH	189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDAT	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2 ⁽¹⁾	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved	18Eh
TMR1H	0Fh	OSCCON	8Fh	EEADRH	10Fh	Reserved	18Fh
T1CON	10h	OSCTUNE	90h		110h		190h
TMR2	11h	SSPCON2	91h		111h		191h
T2CON	12h	PR2	92h		112h		192h
SSPBUF	13h	SSPADD	93h		113h		193h
SSPCON	14h	SSPSTAT	94h		114h		194h
CCPR1L	15h	WPUB	95h		115h		195h
CCPR1H	16h	IOCB	96h	General Purpose Registers	116h	General Purpose Registers	196h
CCP1CON	17h	VRCON	97h		117h		197h
RCSTA	18h	TXSTA	98h	16 Bytes	118h	16 Bytes	198h
TXREG	19h	SPBRG	99h		119h		199h
RCREG	1Ah	SPBRGH	9Ah		11Ah		19Ah
CCPR2L	1Bh	PWM1CON	9Bh		11Bh		19Bh
CCPR2H	1Ch	ECCPAS	9Ch		11Ch		19Ch
CCP2CON	1Dh	PSTRCON	9Dh		11Dh		19Dh
ADRESH	1Eh	ADRESL	9Eh		11Eh		19Eh
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
	20h		A0h		120h		1A0h
General Purpose Registers	3Fh	General Purpose Registers		General Purpose Registers		General Purpose Registers	
	40h	80 Bytes		80 Bytes		80 Bytes	
96 Bytes	6Fh		EFh		16Fh		1EFh
	70h	accesses	F0h	accesses	170h	accesses	1F0h
	7Fh	70h-7Fh	FFh	70h-7Fh	17Fh	70h-7Fh	1FFh
Bank 0		Bank 1		Bank 2		Bank 3	

■ Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.

Note 2: PIC16F887 only.

Figura 2: Resumen de SFRs

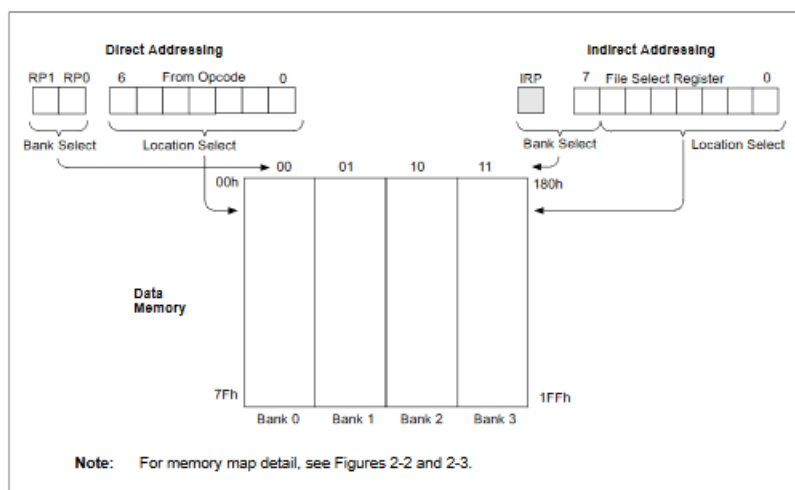


Figura 3: Acceso directo e indirecto

2. Ejercicio 2.2

```

; *****
; TITLE: EXERCISE 2.1
; AUTHOR: VALENZUELA GABRIEL EMANUEL
; SUBJET: DIGITAL ELECTRONICS II
; INSTITUTE: FEFYN – UNC
; DATE: 03-23-2019
; VERSION: 1.0.0
;
; RESUME:
; THIS PROGRAM ADD TO VALUES ON 0X21 AND 0X22
; AFTER STORE THE RESULT ON 0X23 AND 0X24
; *****

```

```
#include "p16f887.inc"
```

```

; CONFIG1
; __config 0xFFF7
_CONFIG _CONFIG1, _FOSC_EXTRC_CLKOUT & _WDTE_OFF & _PWRTE_OFF & _MCLRE_ON & _CP_OFF & _CPD_OFF &
_BOREN_ON & _IESO_ON & _FCMEN_ON & _LVP_ON
; CONFIG2
; __config 0xFFFF
_CONFIG _CONFIG2, _BOR4V_BOR40V & _WRT_OFF

```

```

ORG 0x0 ;ORIGIN
GOTO PROGRAM ;SAVE THE POSITION 0X04
CLRW ;CLEAR THE WORKING REGISTER
;===== DEFINE =====
SUM_ONE EQU 0X21
SUM_TWO EQU 0X22
RESULT_ONE EQU 0X23
RESULT_TWO EQU 0X24
;===== MAIN PROGRAM =====
PROGRAM:
    MOVWF SUM_ONE ;MOVE THE CONTENT OF THE FILE REGISTER TO THE WORKING REGISTER
    ADDWF SUM_TWO,0 ;ADD THE CONTENT OF THE FILE REGISTER, WITH THE CONTENT OF THE WORKING REGISTER
    AND STORE IT ON THE W REGISTER
    MOVWF RESULT_ONE ;COPY THE CONTENT OF THE WORKING REGISTER ON THE FILE REGISTER
    MOVWF RESULT_TWO ;COPY THE CONTENT OF THE WORKING REGISTER ON THE FILE REGISTER
END

```

Referencias

- [1] PÉREZ, Fernando E. Valdés; ARENY, Ramón Pallás. Microcontroladores: Fundamentos y aplicaciones con PIC. Marcombo, 2007.
- [2] PIC16F882/883/884/886/887 Data Sheet