UNIVERSITY OF ESWATINI

MAIN EXAMINATION, SECOND SEMESTER MAY 2019

FACULTY OF SCIENCE AND ENGINEERING

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

TITLE OF PAPER: MICROCONTROLLERS AND MICROCOMPUTER SYSTEMS

COURSE CODE: EE423

TIME ALLOWED: THREE HOURS

INSTRUCTIONS:

1. There are five questions in this paper. Answer any FOUR questions. Each question carries 25 marks.
2. If you think not enough data has been given in any question, you may assume any reasonable values stating your assumptions in each case.

THIS PAPER SHOULD NOT BE OPENED UNTIL PERMISSION HAS BEEN GIVEN BY THE INVIGILATOR

THIS PAPER CONTAINS SIX (6) PAGES INCLUDING THIS PAGE

QUESTION ONE *(25 marks)*

(a) Draw a typical block diagram of a microprocessor based system facilitated with storing user inputs, interfacing analog inputs, interfacing digital I/O, event counting, timing, handling external interrupts and serial communications.

(5 marks)

(b) (i) Briefly state the function of *ALU*, control unit and few dedicated registers in a typical microprocessor.

(6 marks)

(ii) State the differences between *RISC* and *CISC* systems. (2 marks)

(c) A segment of a program written for 16F84A is shown in Figure-Q1.

```
Loop_1    EQU    011Fh
Loop_2    EQU    021Fh
          ........
          ........
          goto   Loop_2
          ........
          ........
Loop_2    bcf    portB,1
          call   Loop_1
          bsf    portB,1
          ........
          ........
Loop_1    bsf    portA,0
          nop
          bsf    portA,1
          nop
          bcf    portA,0
          bcf    portA,1
          retlw  .50
          btfsc  portB,2
          goto   loop_5
          ........
```

Figure – Q1

(i) Find the memory locations where the subroutine section is written.

(3 marks)

(ii) Tabulate the program counter and *STACK* contents between the execution of the instruction before the *CALL* and the instruction after the *CALL*.

(5 marks)

(d) Write the following information for 16F84A microcontroller:
Program memory size, data memory size, data EEPROM size, instruction width, type of architecture and available on-chip modules.

(4 marks)

QUESTION TWO *(25 marks)*

(a) A 3×4 key pad shown in Figure-Q2(a) is required to be connected to 16F84A microcontroller using PortB.

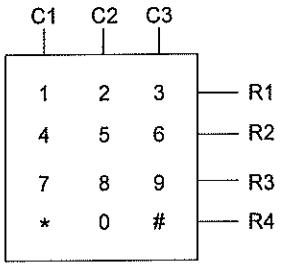


Figure-Q2(a)

```
sub_1   movlw   xx
        movwf   reg_1
loop    nop
        nop
        nop
        decfsz  reg_1,1
        goto    loop
        return
```

Figure-Q2(b)

The keys '#' and '*' may have the decimal values of 10 and 11 respectively.

(i) Draw a diagram showing the interconnections between the keypad and 16F84A.
(4 marks)

(ii) Give the configuration of the relevant register in 16F84A to support the scheme in (i) above.
(3 marks)

(iii) Draw a flow chart segment to show only the scanning of keys $3, 6, 9, \#$ and storing the result in the working register. You may neglect switch bounce.
(6 marks)

(b) In Figure-Q2(b), a subroutine written for 16F84A is shown. The device clock is $10MHz$.

(i) Show how you can limit the execution time of this subroutine as close as $350\mu s$ with minimal changes. What is the percentage timing error in your implementation?
(8 marks)

(ii) Find the maximum possible execution time limit in this subroutine.
(4 marks)

QUESTION THREE *(25 marks)*

An irrigation system for two fields is designed based on 16F84A microcontroller.

A single overhead tank is used to supply water to the two fields through independently controlled gate valves. A pump supplies water to the overhead tank from a well when needed.

Following system components are used and their inputs / outputs are TTL compatible.

Signal	*Function*
T_L	Overhead tank water level sensor output. It is logic '1' if the level is above a set value.
S_1 *and* S_2	Outputs of the two soil moisture sensors of the two fields. Gives logic '0' if the moisture is less than a set value.
V_1 *and* V_2	Control signals for the two gate valves. As long as logic '1' is applied on these lines, the valves remain open.
WP	Control signal of the water pump. As long as logic '1' is applied on this line, the pump keeps running.

(i) Draw a circuit diagram for this system showing all signal interconnections with the pin numbers of 16F84A. Sensors and the controlled components must be shown as blocks.

(5 marks)

(ii) Design a truth table showing the control logic for this system.

(8 marks)

(iii) Draw a flowchart of a program to be used on 16F84A, using (ii) above.

(8 marks)

(iv) Show the configuration instructions of the ports of 16F84A as required.

(4 marks)

QUESTION FOUR (*25 marks*)

(a) Following are with respect to a 16F84A microcontroller program.

(i) The value of $INTCON$ register at the initialization is $B0h$. Explain what you can understand by this setup.

(2 marks)

(ii) When the program is running, the $INTCON$ showed a value of $34h$ at some point. Interpret what is described by this situation.

(2 marks)

(iii) When an interrupt occurs, state the events that will take place in the program counter, stack memory and GIE bit. Indicate the essential features to be included in an interrupt service routine.

(6 marks)

(iv) The program uses $TMR0$ timer to generate a regular interrupt in every 1ms. Show the settings needed in relevant registers to achieve this supported with relevant calculations. What is the value of $INTCON$ register just after such an interrupt? Assume that the system uses a $2MHz$ crystal oscillator.

(6 marks)

(b) An application uses a PIC 16F877 microcontroller with a $10MHz$ crystal oscillator.

(i) It is required to use the Analog to Digital Converter (ADC) of the device with an external voltage reference and five analog input channels. Show the settings of the relevant registers with left justification if the input channel 4 is selected and the ADC is turned on but not started to convert.

(6 marks)

(ii) The ADC minimum acquisition time is about $20\mu s$. State the relevance of this value and indicate how it is incorporated in a program.

(3 marks)

QUESTION FIVE (25 *marks*)

(a) A serial device which supports SPI is required to be connected to the Synchronous Serial Port of a PIC 16F877 microcontroller.

(i) Draw a circuit diagram to show the interconnections of the device and the microcontroller. You may consider the device pins such as DIN $(data\ in)$, $DOUT$ $(data\ out)$, $SCLK$ $(serial\ clock)$ and $\overline{CS}$ $(chip\ select)$. Mark only the relevant pin numbers of the microcontroller used for the interconnection.

(6 marks)

(ii) State briefly how the data is transferred between the device and the microcontroller using SPI mode.

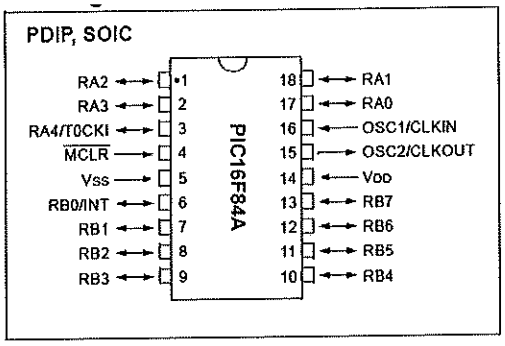
(9 marks)

(b) The serial device has a maximum data rate of $250kHz$ and the microcontroller is running on a $10MHz$ crystal oscillator.

Show the settings of the $SSPCON$, $SSPSTAT$ and any other register involved to establish SPI mode of operation, giving justifications where necessary. Among other typical settings, you may include the settings for, the high idle state of clock, data sampling at the middle of the bit period and the data transfers are on the falling edge of the clock.

(10 marks)

PIC 16F84A



File Address			File Address
00h	Indirect addr.(1)	Indirect addr.(1)	80h
01h	TMR0	OPTION_REG	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h	—	—	87h
08h	EEDATA	EECON1	88h
09h	EEADR	EECON2(1)	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch	68 General Purpose Registers (SRAM)	Mapped (accesses) in Bank 0	8Ch
4Fh			CFh
50h			D0h

STATUS REGISTER (ADDRESS 03h, 83h)

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
bit 7							bit 0

Unimplemented: Maintain as '0'

RP0: Register Bank Select bits (used for direct addressing)
01 = Bank 1 (80h - FFh)
00 = Bank 0 (00h - 7Fh)

$\overline{TO}$: Time-out bit
1 = After power-up, CLRWDT instruction, or SLEEP instruction
0 = A WDT time-out occurred

$\overline{PD}$: Power-down bit
1 = After power-up or by the CLRWDT instruction
0 = By execution of the SLEEP instruction

Z: Zero bit
1 = The result of an arithmetic or logic operation is zero
0 = The result of an arithmetic or logic operation is not zero

DC: Digit carry/$\overline{borrow}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions) (for $\overline{borrow}$, the polarity is reversed)
1 = A carry-out from the 4th low order bit of the result occurred
0 = No carry-out from the 4th low order bit of the result

C: Carry/$\overline{borrow}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions) (for $\overline{borrow}$, the polarity is reversed)
1 = A carry-out from the Most Significant bit of the result occurred
0 = No carry-out from the Most Significant bit of the result occurred

Note: A subtraction is executed by adding the two's complement of the second operand. For rotate (RRF, RLF) instructions, this bit is loaded with either the high or low order bit of the source register.

PIC 16F84A

OPTION REGISTER (ADDRESS 81h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

RBPU: PORTB Pull-up Enable bit
1 = PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual port latch values

INTEDG: Interrupt Edge Select bit
1 = Interrupt on rising edge of RB0/INT pin
0 = Interrupt on falling edge of RB0/INT pin

T0CS: TMR0 Clock Source Select bit
1 = Transition on RA4/T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

T0SE: TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on RA4/T0CKI pin
0 = Increment on low-to-high transition on RA4/T0CKI pin

PSA: Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

PS2:PS0: Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

INTCON REGISTER (ADDRESS 0Bh, 8Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
bit 7							bit 0

GIE: Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts

EEIE: EE Write Complete Interrupt Enable bit
1 = Enables the EE Write Complete interrupts
0 = Disables the EE Write Complete interrupt

T0IE: TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

INTE: RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

RBIE: RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

T0IF: TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

INTF: RB0/INT External Interrupt Flag bit
1 = The RB0/INT external interrupt occurred (must be cleared in software)
0 = The RB0/INT external interrupt did not occur

RBIF: RB Port Change Interrupt Flag bit
1 = At least one of the RB7:RB4 pins changed state (must be cleared in software)
0 = None of the RB7:RB4 pins have changed state

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS						
ADDWF	f, d	Add W and f	1	00 0111 dfff ffff	C,DC,Z	1,2
ANDWF	f, d	AND W with f	1	00 0101 dfff ffff	Z	1,2
CLRF	f	Clear f	1	00 0001 lfff ffff	Z	2
CLRW	-	Clear W	1	00 0001 0xxx xxxx	Z	
COMF	f, d	Complement f	1	00 1001 dfff ffff	Z	1,2
DECF	f, d	Decrement f	1	00 0011 dfff ffff	Z	1,2
DECFSZ	f, d	Decrement f, Skip if 0	1 (2)	00 1011 dfff ffff		1,2,3
INCF	f, d	Increment f	1	00 1010 dfff ffff	Z	1,2
INCFSZ	f, d	Increment f, Skip if 0	1 (2)	00 1111 dfff ffff		1,2,3
IORWF	f, d	Inclusive OR W with f	1	00 0100 dfff ffff	Z	1,2
MOVF	f, d	Move f	1	00 1000 dfff ffff	Z	1,2
MOVWF	f	Move W to f	1	00 0000 lfff ffff		
NOP	-	No Operation	1	00 0000 0xx0 0000		
RLF	f, d	Rotate Left f through Carry	1	00 1101 dfff ffff	C	1,2
RRF	f, d	Rotate Right f through Carry	1	00 1100 dfff ffff	C	1,2
SUBWF	f, d	Subtract W from f	1	00 0010 dfff ffff	C,DC,Z	1,2
SWAPF	f, d	Swap nibbles in f	1	00 1110 dfff ffff		1,2
XORWF	f, d	Exclusive OR W with f	1	00 0110 dfff ffff	Z	1,2
BIT-ORIENTED FILE REGISTER OPERATIONS						
BCF	f, b	Bit Clear f	1	01 00bb bfff ffff		1,2
BSF	f, b	Bit Set f	1	01 01bb bfff ffff		1,2
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01 10bb bfff ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01 11bb bfff ffff		3
LITERAL AND CONTROL OPERATIONS						
ADDLW	k	Add literal and W	1	11 111x kkkk kkkk	C,DC,Z	
ANDLW	k	AND literal with W	1	11 1001 kkkk kkkk	Z	
CALL	k	Call subroutine	2	10 0kkk kkkk kkkk		
CLRWDT	-	Clear Watchdog Timer	1	00 0000 0110 0100	$\overline{TO}$,$\overline{PD}$	
GOTO	k	Go to address	2	10 1kkk kkkk kkkk		
IORLW	k	Inclusive OR literal with W	1	11 1000 kkkk kkkk	Z	
MOVLW	k	Move literal to W	1	11 00xx kkkk kkkk		
RETFIE	-	Return from interrupt	2	00 0000 0000 1001		
RETLW	k	Return with literal in W	2	11 01xx kkkk kkkk		
RETURN	-	Return from Subroutine	2	00 0000 0000 1000		
SLEEP	-	Go into standby mode	1	00 0000 0110 0011	$\overline{TO}$,$\overline{PD}$	
SUBLW	k	Subtract W from literal	1	11 110x kkkk kkkk	C,DC,Z	
XORLW	k	Exclusive OR literal with W	1	11 1010 kkkk kkkk	Z	

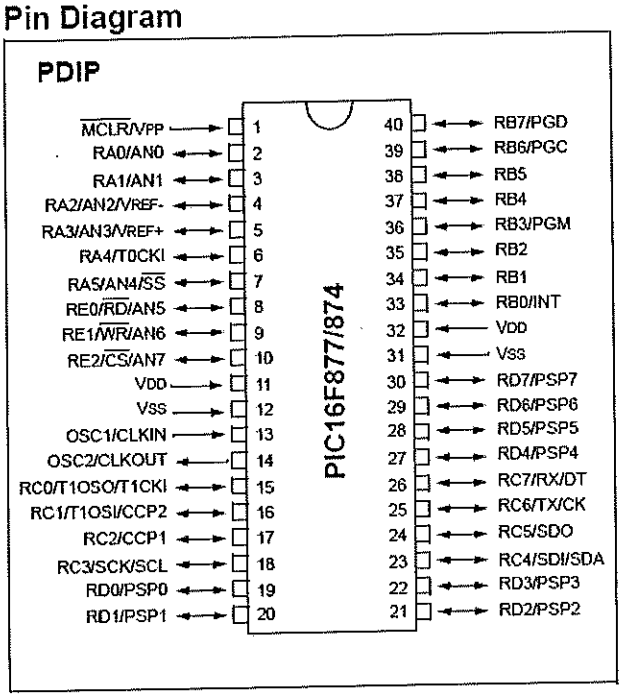
Note 1: When an I/O register is modified as a function of itself (e.g., `MOVF PORTB, 1`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 Module.

3: If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a `NOP`.

PIC 16F877

Pin Diagram



SSPSTAT: SYNC SERIAL PORT STATUS REGISTER (ADDRESS: 94h)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
bit 7							bit 0

SMP: Sample bit
SPI Master mode:
1 = Input data sampled at end of data output time
0 = Input data sampled at middle of data output time
SPI Slave mode:
SMP must be cleared when SPI is used in slave mode
In I^2C Master or Slave mode:
1 = Slew rate control disabled for standard speed mode (100 kHz and 1 MHz)
0 = Slew rate control enabled for high speed mode (400 kHz)

CKE: SPI Clock Edge Select (Figure 9-2, Figure 9-3 and Figure 9-4)
SPI mode:
For CKP = 0
1 = Data transmitted on rising edge of SCK
0 = Data transmitted on falling edge of SCK
For CKP = 1
1 = Data transmitted on falling edge of SCK
0 = Data transmitted on rising edge of SCK
In I^2C Master or Slave mode:
1 = Input levels conform to SMBus spec
0 = Input levels conform to I^2C specs

D/A: Data/$\overline{\text{Address}}$ bit (I^2C mode only)
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

P: STOP bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a STOP bit has been detected last (this bit is '0' on RESET)
0 = STOP bit was not detected last

S: START bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a START bit has been detected last (this bit is '0' on RESET)
0 = START bit was not detected last

R/W: Read/Write bit Information (I^2C mode only)
This bit holds the R/W bit information following the last address match. This bit is only valid from the address match to the next START bit, STOP bit or not ACK bit.
In I^2C Slave mode:
1 = Read
0 = Write
In I^2C Master mode:
1 = Transmit is in progress
0 = Transmit is not in progress
Logical OR of this bit with SEN, RSEN, PEN, RCEN, or ACKEN will indicate if the MSSP is in IDLE mode.

UA: Update Address (10-bit I^2C mode only)
1 = Indicates that the user needs to update the address in the SSPADD register
0 = Address does not need to be updated

BF: Buffer Full Status bit
Receive (SPI and I^2C modes):
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty
Transmit (I^2C mode only):
1 = Data transmit in progress (does not include the ACK and STOP bits), SSPBUF is full
0 = Data transmit complete (does not include the ACK and STOP bits), SSPBUF is empty

SSPCON: SYNC SERIAL PORT CONTROL REGISTER (ADDRESS 14h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
bit 7							bit 0

WCOL: Write Collision Detect bit
Master mode:
1 = A write to SSPBUF was attempted while the I2C conditions were not valid
0 = No collision
Slave mode:
1 = SSPBUF register is written while still transmitting the previous word (must be cleared in software)
0 = No collision

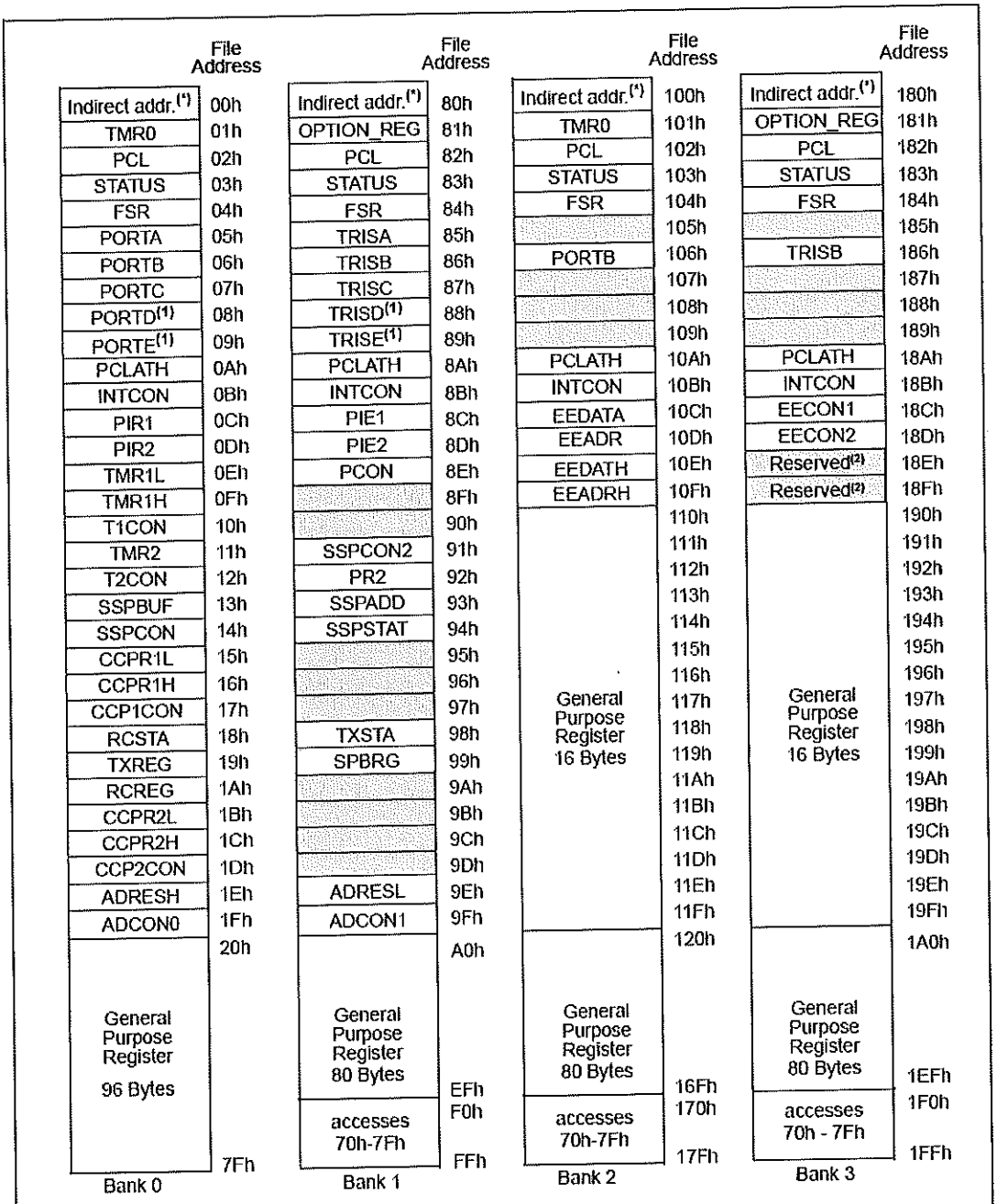
SSPOV: Receive Overflow Indicator bit
In SPI mode:
1 = A new byte is received while SSPBUF holds previous data. Data in SSPSR is lost on overflow. In Slave mode, the user must read the SSPBUF, even if only transmitting data, to avoid overflows. In Master mode, the overflow bit is not set, since each operation is initiated by writing to the SSPBUF register. (Must be cleared in software.)
0 = No overflow
In I^2C mode:
1 = A byte is received while the SSPBUF is holding the previous byte. SSPOV is a "don't care" in Transmit mode. (Must be cleared in software.)
0 = No overflow

SSPEN: Synchronous Serial Port Enable bit
In SPI mode,
When enabled, these pins must be properly configured as input or output
1 = Enables serial port and configures SCK, SDO, SDI, and SS as the source of the serial port pins
0 = Disables serial port and configures these pins as I/O port pins
In I^2C mode,
When enabled, these pins must be properly configured as input or output
1 = Enables the serial port and configures the SDA and SCL pins as the source of the serial port pins
0 = Disables serial port and configures these pins as I/O port pins

CKP: Clock Polarity Select bit
In SPI mode:
1 = Idle state for clock is a high level
0 = Idle state for clock is a low level
In I^2C Slave mode:
SCK release control
1 = Enable clock
0 = Holds clock low (clock stretch). (Used to ensure data setup time.)
In I^2C Master mode:
Unused in this mode

SSPM3:SSPM0: Synchronous Serial Port Mode Select bits
0000 = SPI Master mode, clock = Fosc/4
0001 = SPI Master mode, clock = Fosc/16
0010 = SPI Master mode, clock = Fosc/64
0011 = SPI Master mode, clock = TMR2 output/2
0100 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control enabled.
0101 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control disabled. $\overline{SS}$ can be used as I/O pin.
0110 = I^2C Slave mode, 7-bit address
0111 = I^2C Slave mode, 10-bit address
1000 = I^2C Master mode, clock = Fosc / (4 * (SSPADD+1))
1011 = I^2C Firmware Controlled Master mode (slave idle)
1110 = I^2C Firmware Controlled Master mode, 7-bit address with START and STOP bit interrupts enabled
1111 = I^2C Firmware Controlled Master mode, 10-bit address with START and STOP bit interrupts enabled
1001, 1010, 1100, 1101 = Reserved

PIC16F877/876 REGISTER FILE MAP



Unimplemented data memory locations, read as '0'.
* Not a physical register.

Note 1: These registers are not implemented on the PIC16F876.
2: These registers are reserved, maintain these registers clear.

ADCON0 REGISTER (ADDRESS: 1Fh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	—	ADON
bit 7							bit 0

ADCS1:ADCS0: A/D Conversion Clock Select bits
00 = Fosc/2
01 = Fosc/8
10 = Fosc/32
11 = FRC (clock derived from the internal A/D module RC oscillator)

CHS2:CHS0: Analog Channel Select bits
000 = channel 0, (RA0/AN0)
001 = channel 1, (RA1/AN1)
010 = channel 2, (RA2/AN2)
011 = channel 3, (RA3/AN3)
100 = channel 4, (RA5/AN4)
101 = channel 5, (RE0/AN5)(1)
110 = channel 6, (RE1/AN6)(1)
111 = channel 7, (RE2/AN7)(1)

GO/$\overline{\text{DONE}}$: A/D Conversion Status bit
If ADON = 1:
1 = A/D conversion in progress (setting this bit starts the A/D conversion)
0 = A/D conversion not in progress (this bit is automatically cleared by hardware when the A/D conversion is complete)

Unimplemented: Read as '0'

ADON: A/D On bit
1 = A/D converter module is operating
0 = A/D converter module is shut-off and consumes no operating current

ADCON1 REGISTER (ADDRESS 9Fh)

U-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

ADFM: A/D Result Format Select bit
1 = Right justified. 6 Most Significant bits of ADRESH are read as '0'.
0 = Left justified. 6 Least Significant bits of ADRESL are read as '0'.

Unimplemented: Read as '0'

PCFG3:PCFG0: A/D Port Configuration Control bits:

PCFG3: PCFG0	AN7(1) RE2	AN6(1) RE1	AN5(1) RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF+	VREF-	CHAN/ Refs(2)
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	RA3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	RA3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	RA3	VSS	2/1
011x	D	D	D	D	D	D	D	D	VDD	VSS	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	RA3	RA2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	RA3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	RA3	RA2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	RA3	RA2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	RA3	RA2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	RA3	RA2	1/2

A = Analog input D = Digital I/O

Note 1: These channels are not available on PIC16F873/876 devices.
2: This column indicates the number of analog channels available as A/D inputs and the number of analog channels used as voltage reference inputs.

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
- n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

FIGURE 10-1: USART TRANSMIT BLOCK DIAGRAM

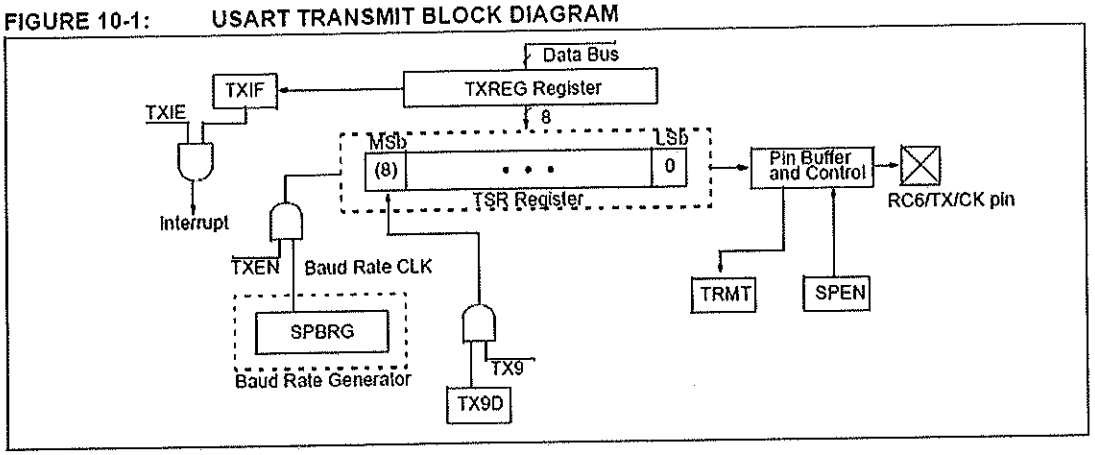
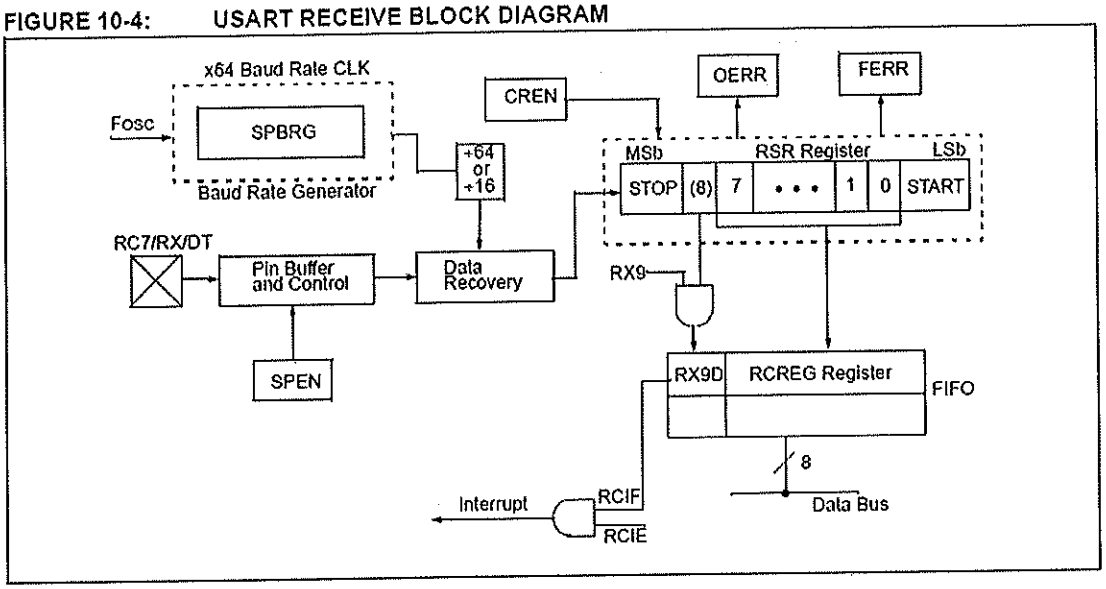


FIGURE 10-4: USART RECEIVE BLOCK DIAGRAM



TXSTA:

CSRC	TX9	TXEN	SYNC	-	BRGH	TRMT	TX9D

RCSTA:

SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RXPD

$$BR_H = \frac{F_{osc}}{16(X+1)} \qquad BR_L = \frac{F_{osc}}{64(X+1)}$$

SPI Mode Waveform (Master Mode)