UNIVERSITI TUN HUSSEIN ONN MALAYSIA

FINAL EXAMINATION
SEMESTER II
SESSION 2008/2009

COURSE NAME	: MICROPROCESSOR AND MICROCONTROLLER
COURSE CODE	: BEE 3233
PROGRAMME	: 3 BEE
DATE OF EXAMINATION	: APRIL 2009 / MAY 2009
DURATION	: 3 HOURS
INSTRUCTION	: ANSWER ANY FIVE (5) QUESTIONS FROM SEVEN (7) QUESTIONS

THIS QUESTION PAPER CONTAINS SIXTEEN (16) PAGES

Q1 (a) Programmer's Model of 68000 comprises of many CPU registers. With the aid of diagram, explain briefly the functions for each component in Programmer's Model of 68000.

(7 marks)

(b) Motorola 68000 is a microprocessor that has been used in Macintosh computer. Explain briefly four advantages of the 68000 microprocessor.

(4 marks)

(c) CPU and its memory communicate via the system bus. Determine the maximum memory size for 68000 and calculate the address range to access memory.

(3 marks)

(d) List CCR flags in 68000 microprocessor and describe each function. If the Status Register value is 2506, what is the flag bit for C, Carry.

(3 marks)

Q2 (a) Absolute Addressing Mode used to fetch data in memory. The two variations of absolute mode are Absolute Long Mode and Absolute Short Mode. Explain the difference, function and advantage between these two addressing mode.

(6 marks)

(b) Given D0=$0000 FFFF, D1 = $0000 7FFF and status register CCR is XNZVC = 11111. All instructions are independent. What is the value of CCR after executing these instructions?

(i) ADD.L #1,D0
(ii) SUB.B #$7F,D1
(iii) MULS D0,D1
(iv) ROXR.B D1

(4 marks)

(c) Write the appropriate logic instruction to preserve bits 0, 3-9 and 13 of register D2, and clear all others.

(2 marks)

(d) Change the C program below to the assembly language where x is located in memory address $2000 and y is located in memory address $3000. Then, store the result for x in data register, D0 and the result for y in D3.

```
if (y>x)
        y = y+x
else if (y<x)
        x = x-y
else x = y
```

(8 marks)

Q3 (a) Explain the different functions of the internal A0 bit when used for:
(i) byte addressing
(ii) word addressing

(4 marks)

(b) Show the state of UDS and LDS when the 68000 is involved in the following memory accesses:
(i) a byte write to address 3000
(ii) a byte write to address 3001
(iii) a word write to address 3000

(6 marks)

(c) Design partial decoder for a system with a ROM device which is 16k word (32k byte) starting address from $000000, RAM device which is 64k word (128k byte) starting address from $400000 and one Input/Output device place at address $800000 until $80001F (32 addresses).

(10 marks)

Q4 (a) Briefly explain the type of architecture used in the PIC microcontroller. What are its advantages ?

(4 marks)

(b) Explain the memory organisation of the PIC16F84A microcontroller as regards,
(i) Program Memory
(ii) Data memory

(8 marks)

(c) Explain the function of PORTA and TRISA register in the PIC16F84A microcontroller

(4 marks)

(d) Write an assembly language program to set PORT RA0, 1, 2, 3 as inputs and RA4 as output.

(4 marks)

Q5 (a) Explain the advantages of macro compare to subroutine.

(5 marks)

(b) Explain the concept of stack in the PIC16F84A microcontroller.

(5 marks)

(c) PIC 16F84 does not have a division instruction. Therefore, write a macro to implement an instruction called DIVLW which will divide an 8-bit unsigned number in working register, WREG by a 4-bit unsigned literal number. The division product is a 4-bit number stored in lower four bit of working register and the remainder is stored in upper four bit of working register. Example of using this macro is shown below:

```
MOVLW   15h     ;load WREG with 15h
DIVLW   04h     ;divide 15h in WREG by literal value
                ;04h
                ;Result is a 4 bit unsigned number
                 in lower 4 bit of WREG and remainder
                 Is in upper 4 bit of WREG
```

(10 marks)

Q6 (a) State the number of clock cycles in a PIC instruction cycle and the number of instruction cycles taken to execute the following instructions:

(i) ADDLW 50h
(ii) RRF PORTB
(iii) RETLW 24h
(iv) IORWF FEh

(4 marks)

(b) What is difference between control directives DEFINE and EQU? Give a related example for each of them.

(4 marks)

(c) Calculate the pre-load value required in TMR0 register to obtain a delay of 4ms between the load operation and the T0IF going high, if the clock rate is 4MHz and the value of PS2:PS1:PS0 in OPTION register are 1:0:0.

(4 marks)

(d) PIC16F84 is used to execute subroutine 1. If PIC clock input is 200kHz, calculate total delay to execute the subroutine1. (Initial C=0).

```
subroutine1
                MOVLW   10h
                MOVWF   20h
                MOVLW   .2
                MOVWF   30h
                MOVLW   .1
LOOP            RRF     20h
                SUBWF   30h
                BTFSS   STATUS,2
                GOTO    LOOP

                RETURN
```

(8 marks)

Q7 (a) The A/D converter has one 10-bit output, determine the resolution per bit and calculate the equivalent digital value of the output for the input given below (reference voltage, *Vref* = 5V):

i) 1 V
ii) 1.25 V
iii) 3.45 V

(4 marks)

(b) Write a sequence of instructions to initialize the analog to digital (A/D) converter in PIC16C71 if:

- PIC16C71 is clocked at 4 MHz
- A/D clock is $F_{osc}/8$
- Analog input is on RA0

(4 marks)

(c) Figure Q7(c) shows a temperature control system using PIC16C71 microcontroller. A temperature sensor is used to send the temperature in analog and to provide feedback to the microcontroller. Based on this feedback, the microcontroller controls the heater and the cooler in order to achieve the desired temperature in the plant.

Assume that the desired temperature is between 25°C and 30°C and the value in digital for 25°C is decimal 25 and 30°C is decimal 30.

From the statement above, answer the questions below:

i) Sketch the microcontroller circuit for the system and label the input and output port that you want to use in the system if:
   - The system used one analog channel
   - The input for the driver is four bits
   - Temperature sensor used is LM35

   Your circuit should include:
   - Reset circuit
   - Clock circuit
   - Connection between sensor and PIC
   - Connection between PIC and driver

ii) Analyze and plan the program for the system by using flowchart.

(12 marks)

FINAL EXAMINATION

SEMESTER/SESSION : II/2008/09
COURSE : MICROPROCESSOR AND MICROCONTROLLER

PROGRAMME : 3 BEE
COURSE CODE : BEE 3233

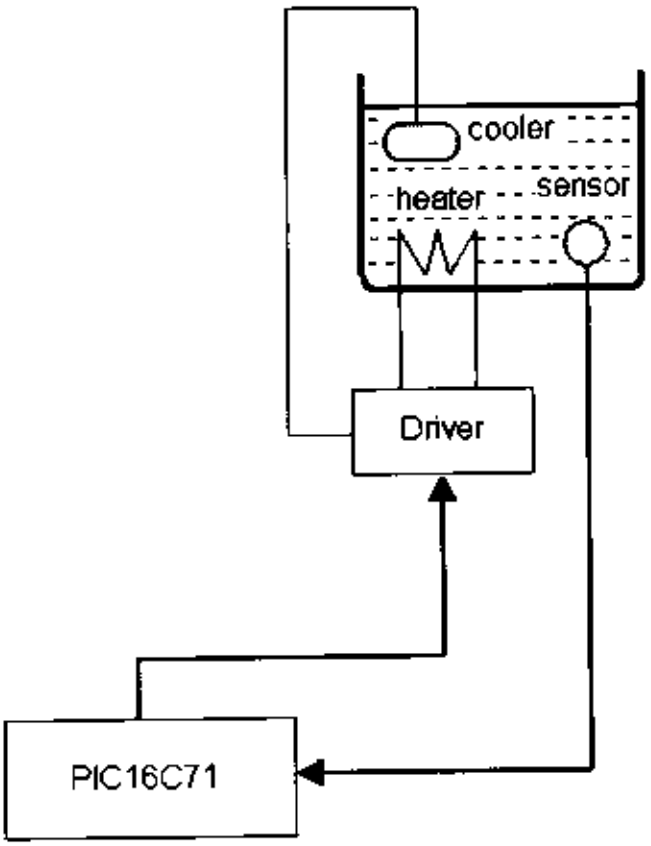

Figure Q7(c)

FINAL EXAMINATION

SEMESTER/SESSION : II/2008/09
COURSE : MICROPROCESSOR AND MICROCONTROLLER

PROGRAMME : 3 BEE
COURSE CODE : BEE 3233

Special Function Register (SFR) File Summary for PIC16F84A

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on RESET	Details on page
Bank 0											
00h	INDF	Uses contents of FSR to address Data Memory (not a physical register)								---- ----	11
01h	TMR0	8-bit Real-Time Clock/Counter								xxxx xxxx	20
02h	PCL	Low Order 8 bits of the Program Counter (PC)								0000 0000	11
03h	STATUS(2)	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	8
04h	FSR	Indirect Data Memory Address Pointer 0								xxxx xxxx	11
05h	PORTA(4)	—	—	—	RA4/T0CKI	RA3	RA2	RA1	RA0	---x xxxx	16
06h	PORTB(5)	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0/INT	xxxx xxxx	18
07h	—	Unimplemented location, read as '0'								—	—
08h	EEDATA	EEPROM Data Register								xxxx xxxx	13,14
09h	EEADR	EEPROM Address Register								xxxx xxxx	13,14
0Ah	PCLATH	—	—	—	Write Buffer for upper 5 bits of the PC(1)					---0 0000	11
0Bh	INTCON	GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	10
Bank 1											
80h	INDF	Uses Contents of FSR to address Data Memory (not a physical register)								---- ----	11
81h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	9
82h	PCL	Low order 8 bits of Program Counter (PC)								0000 0000	11
83h	STATUS (2)	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	8
84h	FSR	Indirect data memory address pointer 0								xxxx xxxx	11
85h	TRISA	—	—	—	PORTA Data Direction Register					---1 1111	16
86h	TRISB	PORTB Data Direction Register								1111 1111	18
87h	—	Unimplemented location, read as '0'								—	—
88h	EECON1	—	—	—	EEIF	WRERR	WREN	WR	RD	---0 x000	13
89h	EECON2	EEPROM Control Register 2 (not a physical register)								---- ----	14
0Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of the PC(1)					---0 0000	11
0Bh	INTCON	GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	10

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0', q = value depends on condition

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a slave register for PC<12:8>. The contents of PCLATH can be transferred to the upper byte of the program counter, but the contents of PC<12:8> are never transferred to PCLATH.

2: The $\overline{TO}$ and $\overline{PD}$ status bits in the STATUS register are not affected by a $\overline{MCLR}$ Reset.

3: Other (non power-up) RESETS include: external RESET through $\overline{MCLR}$ and the Watchdog Timer Reset.

4: On any device RESET, these pins are configured as inputs.

5: This is the value that will be in the port output latch.

FINAL EXAMINATION

SEMESTER/SESSION : II/2008/09
COURSE : MICROPROCESSOR AND MICROCONTROLLER

PROGRAMME : 3 BEE
COURSE CODE : BEE 3233

STATUS Register of PIC16F84A

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

bit 7-6 **Unimplemented:** Maintain as '0'

bit 5 **RP0**: Register Bank Select bits (used for direct addressing)
01 = Bank 1 (80h - FFh)
00 = Bank 0 (00h - 7Fh)

bit 4 **$\overline{TO}$**: Time-out bit
1 = After power-up, CLRWDT instruction, or SLEEP instruction
0 = A WDT time-out occurred

bit 3 **$\overline{PD}$**: Power-down bit
1 = After power-up or by the CLRWDT instruction
0 = By execution of the SLEEP instruction

bit 2 **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

bit 1 **DC**: Digit carry/$\overline{\text{borrow}}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions) (for $\overline{\text{borrow}}$, 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

bit 0 **C**: Carry/$\overline{\text{borrow}}$ bit (ADDWF, ADDLW, SUBLW, SUBWF Instructions) (for $\overline{\text{borrow}}$, 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.

FINAL EXAMINATION

SEMESTER/SESSION : II/2008/09
COURSE : MICROPROCESSOR AND MICROCONTROLLER

PROGRAMME : 3 BEE
COURSE CODE : BEE 3233

OPTION Register of PIC16F84A / PIC16C71

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

bit 7 **$\overline{\text{RBPU}}$**: PORTB Pull-up Enable bit
1 = PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual port latch values

bit 6 **INTEDG**: Interrupt Edge Select bit
1 = Interrupt on rising edge of RB0/INT pin
0 = Interrupt on falling edge of RB0/INT pin

bit 5 **T0CS**: TMR0 Clock Source Select bit
1 = Transition on RA4/T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4 **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

bit 3 **PSA**: Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

bit 2-0 **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

FINAL EXAMINATION

SEMESTER/SESSION : II/2008/09
COURSE : MICROPROCESSOR AND MICROCONTROLLER

PROGRAMME : 3 BEE
COURSE CODE : BEE 3233

INTCON Register of PIC16F84A

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

bit 7 **GIE**: Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts

bit 6 **EEIE**: EE Write Complete Interrupt Enable bit
1 = Enables the EE Write Complete interrupts
0 = Disables the EE Write Complete interrupt

bit 5 **T0IE**: TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4 **INTE**: RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

bit 3 **RBIE**: RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2 **T0IF**: TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1 **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

bit 0 **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

FINAL EXAMINATION

SEMESTER/SESSION : II/2008/09
COURSE : MICROPROCESSOR AND MICROCONTROLLER

PROGRAMME : 3 BEE
COURSE CODE : BEE 3233

Special Function Register (SFR) File Summary for PIC16C71

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (1)
Bank 0											
00h(3)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
02h(3)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h(3)	STATUS	IRP(5)	RP1(5)	RP0	TO	PD	Z	DC	C	0001 1xxx	000q quuu
04h(3)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	—	PORTA Data Latch when written: PORTA pins when read					---x 0000	---u 0000
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
07h	—	Unimplemented								—	—
08h	ADCON0	ADCS1	ADCS0	(6)	CHS1	CHS0	GO/DONE	ADIF	ADON	00-0 0000	00-0 0000
09h(3)	ADRES	A/D Result Register								xxxx xxxx	uuuu uuuu
0Ah(2,3)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh(3)	INTCON	GIE	ADIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
Bank 1											
80h(3)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h(3)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h(3)	STATUS	IRP(5)	RP1(5)	RP0	TO	PD	Z	DC	C	0001 1xxx	000q quuu
84h(3)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	—	PORTA Data Direction Register					---1 1111	---1 1111
86h	TRISB	PORTB Data Direction Control Register								1111 1111	1111 1111
87h(4)	PCON	—	—	—	—	—	—	POR	BOR	---- --qq	---- --uu
88h	ADCON1	—	—	—	—	—	—	PCFG1	PCFG0	---- --00	---- --00
89h(3)	ADRES	A/D Result Register								xxxx xxxx	uuuu uuuu
8Ah(2,3)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh(3)	INTCON	GIE	ADIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0'. Shaded locations are unimplemented, read as '0'

Note 1: Other (non power-up) resets include external reset through $\overline{MCLR}$ and Watchdog Timer Reset
2: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.
3: These registers can be addressed from either bank.
4: The PCON register is not physically implemented in the PIC16C71. read as '0'
5: The IRP and RP1 bits are reserved on the PIC16C710/71/711, always maintain these bits clear
6: Bit5 of ADCON0 is a General Purpose R/W bit for the PIC16C710/711 only. For the PIC16C71, this bit is unimplemented, read as '0'.

FINAL EXAMINATION

SEMESTER/SESSION : II/2008/09
COURSE : MICROPROCESSOR AND MICROCONTROLLER

PROGRAMME : 3 BEE
COURSE CODE : BEE 3233

STATUS Register of PIC16C71

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
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **IRP**: Register Bank Select bit (used for indirect addressing)
1 = Bank 2, 3 (100h - 1FFh)
0 = Bank 0, 1 (00h - FFh)

bit 6-5: **RP1:RP0**: Register Bank Select bits (used for direct addressing)
11 = Bank 3 (180h - 1FFh)
10 = Bank 2 (100h - 17Fh)
01 = Bank 1 (80h - FFh)
00 = Bank 0 (00h - 7Fh)
Each bank is 128 bytes

bit 4: $\overline{TO}$: Time-out bit
1 = After power-up, CLRWDT instruction, or SLEEP instruction
0 = A WDT time-out occurred

bit 3: $\overline{PD}$: Power-down bit
1 = After power-up or by the CLRWDT instruction
0 = By execution of the SLEEP instruction

bit 2: **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

bit 1: **DC**: Digit carry/$\overline{\text{borrow}}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)(for $\overline{\text{borrow}}$ 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

bit 0: **C**: Carry/$\overline{\text{borrow}}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)
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: For $\overline{\text{borrow}}$ the polarity is reversed. 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.

FINAL EXAMINATION

SEMESTER/SESSION : II/2008/09
COURSE : MICROPROCESSOR AND MICROCONTROLLER

PROGRAMME : 3 BEE
COURSE CODE : BEE 3233

INTCON Register of PIC16C71

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	ADIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **GIE:**(1) Global Interrupt Enable bit
1 = Enables all un-masked interrupts
0 = Disables all interrupts

bit 6: **ADIE:** A/D Converter Interrupt Enable bit
1 = Enables A/D interrupt
0 = Disables A/D interrupt

bit 5: **T0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4: **INTE:** RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

bit 3: **RBIE:** RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2: **T0IF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1: **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

bit 0: **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

FINAL EXAMINATION

SEMESTER/SESSION : II/2008/09
COURSE : MICROPROCESSOR AND MICROCONTROLLER

PROGRAMME : 3 BEE
COURSE CODE : BEE 3233

ADCON0 Register of PIC16C71

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADCS1	ADCS0	— (1)	CHS1	CHS0	GO/DONE	ADIF	ADON
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7-6: **ADCS1:ADCS0:** A/D Conversion Clock Select bits
00 = Fosc/2
01 = Fosc/8
10 = Fosc/32
11 = FRC (clock derived from an RC oscillation)

bit 5: **Unimplemented:** Read as '0'.

bit 4-3: **CHS1:CHS0:** Analog Channel Select bits
00 = channel 0, (RA0/AN0)
01 = channel 1, (RA1/AN1)
10 = channel 2, (RA2/AN2)
11 = channel 3, (RA3/AN3)

bit 2: **GO/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)

bit 1: **ADIF:** A/D Conversion Complete Interrupt Flag bit
1 = conversion is complete (must be cleared in software)
0 = conversion is not complete

bit 0: **ADON:** A/D On bit
1 = A/D converter module is operating
0 = A/D converter module is shutoff and consumes no operating current

Note 1: Bit5 of ADCON0 is a General Purpose R/W bit for the PIC16C710/711 only. For the PIC16C71, this bit is unimplemented, read as '0'.

FINAL EXAMINATION

SEMESTER/SESSION : II/2008/09
COURSE : MICROPROCESSOR AND MICROCONTROLLER

PROGRAMME : 3 BEE
COURSE CODE : BEE 3233

ADCON1 Register of PIC16C71

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	PCFG1	PCFG0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7-2: **Unimplemented:** Read as '0'

bit 1-0: **PCFG1:PCFG0**: A/D Port Configuration Control bits

PCFG1:PCFG0	RA1 & RA0	RA2	RA3	VREF
00	A	A	A	VDD
01	A	A	VREF	RA3
10	A	D	D	VDD
11	D	D	D	VDD

A = Analog input

D = Digital I/O

FINAL EXAMINATION

SEMESTER/SESSION : II/2008/09
COURSE : MICROPROCESSOR AND MICROCONTROLLER

PROGRAMME : 3 BEE
COURSE CODE : BEE 3233

PIC16F84A Instruction Set Summary

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	