UNIVERSITI TUN HUSSEIN ONN MALAYSIA

FINAL EXAMINATION
SEMESTER I
SESSION 2010/2011

COURSE	: MICROPROCESSOR AND MICROCONTROLLER
COURSE CODE	: BEE 3233
PROGRAMME	: 3 BEE
EXAMINATION DATE	: NOVEMBER/DECEMBER 2010
DURATION	: 3 HOURS
INSTRUCTIONS	: ANSWER ALL QUESTIONS IN PART A AND ANY TWO (2) QUESTIONS IN PART B

THIS PAPER CONSISTS OF TWENTY FOUR (24) PAGES

PART A

Q1 Figure Q1 shows a simple application using PIC 16C71 to read two analogs input via RA0 and RA1 pins. There are two LEDs connected to RB0 and RB1 respectively. RB2 is connected to another output called output 1. The PIC is clocked at 8MHz.

(a) Determine and explain the initialization value of ADCON0 to implement the circuit in Figure Q1 and suggest the best value for ADCON1.

(5 marks)

(b) Write a sequence of instructions to initialize the input, output and ADC.

(4 marks)

(c) Write a simple program to light the LED1 when the analog value of RA0 is greater than RA1 and light the LED2 when the analog value of RA1 is greater than RA0.

(9 marks)

(d) If the analog value read from RA1 is needed to be sent to a computer using RS232 standard and Figure Q1 (d) shows the program to send the value.

(i) Calculate the duration of stop bit, T and the transmission speed in bit/s.

(5.5 marks)

(ii) Determine the data rate of the transmission.

(1.5 marks)

Q2 (a) Figure Q2 (a) shows the MC68000's assembly program written in EASY68K simulation software. Analyze the program and answer the following questions:

(i) After the program is assembled, determine the first address of the program and the data respectively?

(2 marks)

(ii) After the program is executed, write the value of the data in the Table Q2(a) for the given address.

(6 marks)

(iii) What is content of the CCR (Conditions Code Register) after the microprocessor executes the ADD.W Numbers+4, D0?

(2 marks)

(iv) What are the addressing modes for instruction ADD.W Numbers,D0?

(2 marks)

(b) What is the difference between instruction MOVE.B #50,D0 and MOVEQ #50,D0? Evaluate which instruction to be used if the data is 32-bit and the application has limited memory location to store the program?

(5 marks)

(c) Write a sequence of MC68000 instruction to compare two unsigned byte value located in memory. The first byte is located in $2000, and the second byte is located in $2001. Store the bigger value in address location $2002.

(8 marks)

PART B

Q3 (a) List four (4) interrupt sources in PIC16F84A.

(4 marks)

(b) Given the clock input for the PIC16F84A is 4MHz. Determine the value of prescale value to be used for the TMR0 overflow every 1024μs.

(4 marks)

(c) Suggest and explain the value of OPTION and INTCON register if the specifications are as in Q3 (b).

(8 marks)

(d) If the PIC16F84A is connected to a LED as shown in Figure Q3 (d). Write a simple program to blink the LED every 1024μs using specifications as in Q3 (b).

(7 marks)

(e) Explain the advantage of using TMR0 overflow interrupt as compared to the polling on T0IF flag method.

(2 marks)

Q4 (a) List five (5) Special Function Registers, (SFR) and explain the function of each register that you have listed.

(5 marks)

(b) Write PIC16F84 assembly language based on the statement below:

(i) Select BANK1

(1 mark)

(ii) Initialize the Port if RA1 and RA2 are connected to the switches while RB3, RB4 and RB5 are connected to LEDs.

(2 marks)

(iii) Move value 50H to file register 30H.

(1 mark)

(iv) Move 8-bit of data from file register named REG1 to another file register name REG2.

(1 mark)

(v) Subtract 5 from file register 30H, if the content of 30H not zero, repeat the subtraction.

(2 marks)

(c) A Microprocessor MC68000 is used as a main processor in a mobile robot as shown in Figure Q4(c). The robot has two wheels controlled by two DC motors. To move a robot forward, send $40 to Port B Data Register (PBDR) of PI/T. To move the robot backward, send $80 to Port B Data Register (PBDR) of PI/T. To stop the robot, send $00 to PBDR of PI/T. Given the address of the PBDR is at $800012 and a one second delay subroutine as below.

```
Delay   MOVE.L   #551800,D1
DEL     SUBQ.L   D1
        BNE      DEL
        RTS
```

(i) Write a program to move robot forward for five seconds, then stop for two seconds and then move backward for five seconds.

(10 marks)

(ii) Explain the advantage of using microprocessor (as compare to microcontroller) in developing an application that requires a lot of calculations.

(3 marks)

Q5 (a) CPU and its memory communicate via the system bus. Estimates the maximum memory size for MC68000 and determine the address range to access memory.

(3 marks)

(b) The MC68000 microprocessor contains 16-bit data bus which capable of reading and writing 8-bit and 16-bit of data.

(i) Define the acronym '$\overline{\text{UDS}}$' and '$\overline{\text{LDS}}$'.

(1 marks)

(ii) Explain how are $\overline{\text{UDS}}$ and $\overline{\text{LDS}}$ associated with reading and writing data in the processor.

(2 marks)

(iii) Show the states of $\overline{\text{UDS}}$ and $\overline{\text{LDS}}$ when MC68000 attempts to perform a data access to an odd location.

(2 marks)

(c) Figure Q5(c) shows the address decoding circuit for MC68000 microprocessor.

(i) Based on Figure Q5(c), generate address decoder table and determine the range of address line for each RAM and ROM.

(5 marks)

(ii) Select the suitable size of RAM and EPROM for this system.

(2 marks)

(iii) Draw the memory map for this system.

(3 marks)

(iv) Simplify the design of address decoding circuit in Figure Q5(c) by using partial-address decoder.

(7 marks)

FINAL EXAMINATION

SEMESTER/SESSION : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT CODE : BEE 3233

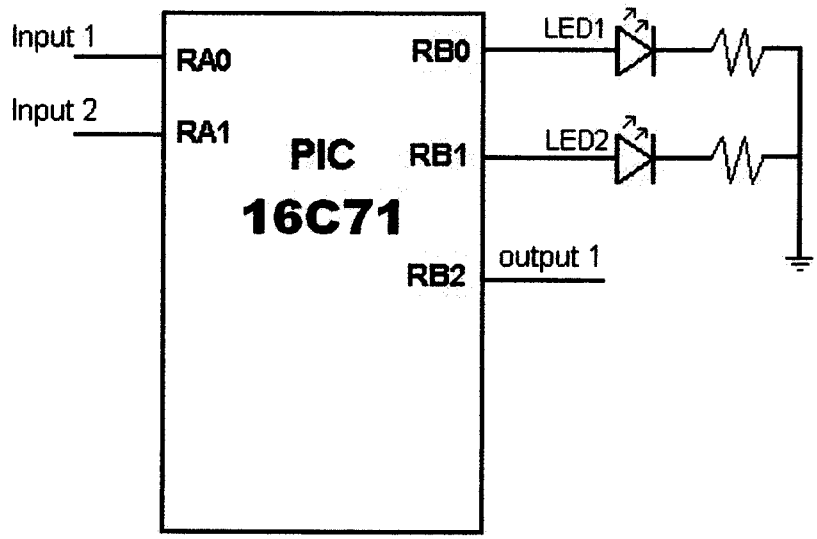


Figure Q1

FINAL EXAMINATION

SEMESTER/SESSION : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT CODE : BEE 3233

```
            MOVF    ADRES,W     ;Copy value of A1 to W
            MOVWF   TXDATA      ;and save to TXDATA
            CALL    SEND        ;Do send routine
            GOTO    $           ;Program wait here

SEND        BCF     PORTB,2     ;Start bit
            CALL    Bitdelay
Local i=0
While i<8                       ;do 8 times starting from LSB
            BTFSC   TXDATA,I
            CALL    SEND1       ;if bit i equal 1, call SEND1
            BTFSS   TXDATA,i
            CALL    SEND0       ;if bit i equal 0, call SEND0
            i=i+1
endw
            BSF     PORTB,2     ;Stop bit
            CALL    Bitdelay
            RETURN

SEND1       BSF     PORTB,2     ;Set line to 1
            CALL    Bitdelay    ;Delay for duration of sending
            RETURN              ;1 bit

SEND0       BCF     PORTB,2     ;Clear line to 0
            CALL    Bitdelay    ;Delay for duration of sending
            RETURN              ;1 bit

Bitdelay    NOP
            MOVLW   0x20        ;Value to get a delay of ????
            MOVWF   COUNTER     ;total delay between 2 bits
LOOP        DECFSZ  COUNTER
            GOTO    LOOP
            RETURN
```

Figure Q1(d)

FINAL EXAMINATION

SEMESTER/SESSION : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT CODE : BEE 3233

```
CODE        EQU         $1000
DATA        EQU         $2000

            ORG         CODE
PROG        MOVE.W      NUMBERS,D0
            ADD.W       NUMBERS+2,D0
            ADD.W       NUMBERS+4,D0
            MOVE.W      D0,SUM

            ORG         DATA
NUMBERS     DC.W        $0011
            DC.W        $0022
            DC.W        $0033
SUM         DC.W        0
            END         PROG
```

Figure Q2(a)

Table Q2(a)

Address	2000	2002	2004	2006
Content				

FINAL EXAMINATION

SEMESTER/SESSION : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT CODE : BEE 3233

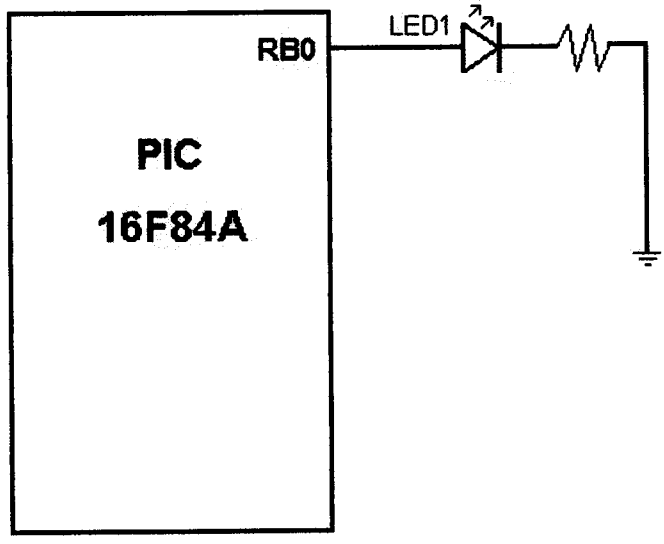


Figure Q3(d)

FINAL EXAMINATION

SEMESTER/SESSION : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT CODE : BEE 3233

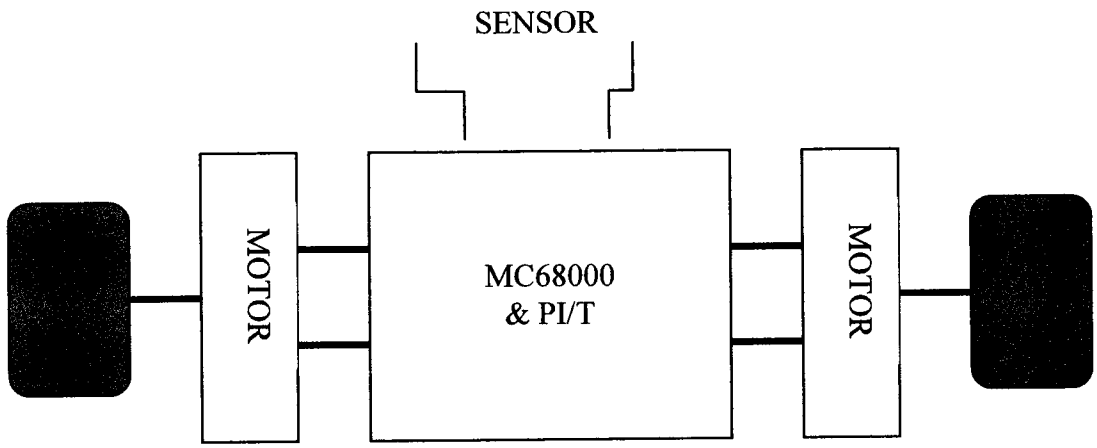


Figure Q4(c)

FINAL EXAMINATION

SEMESTER/SESSION : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT CODE : BEE 3233

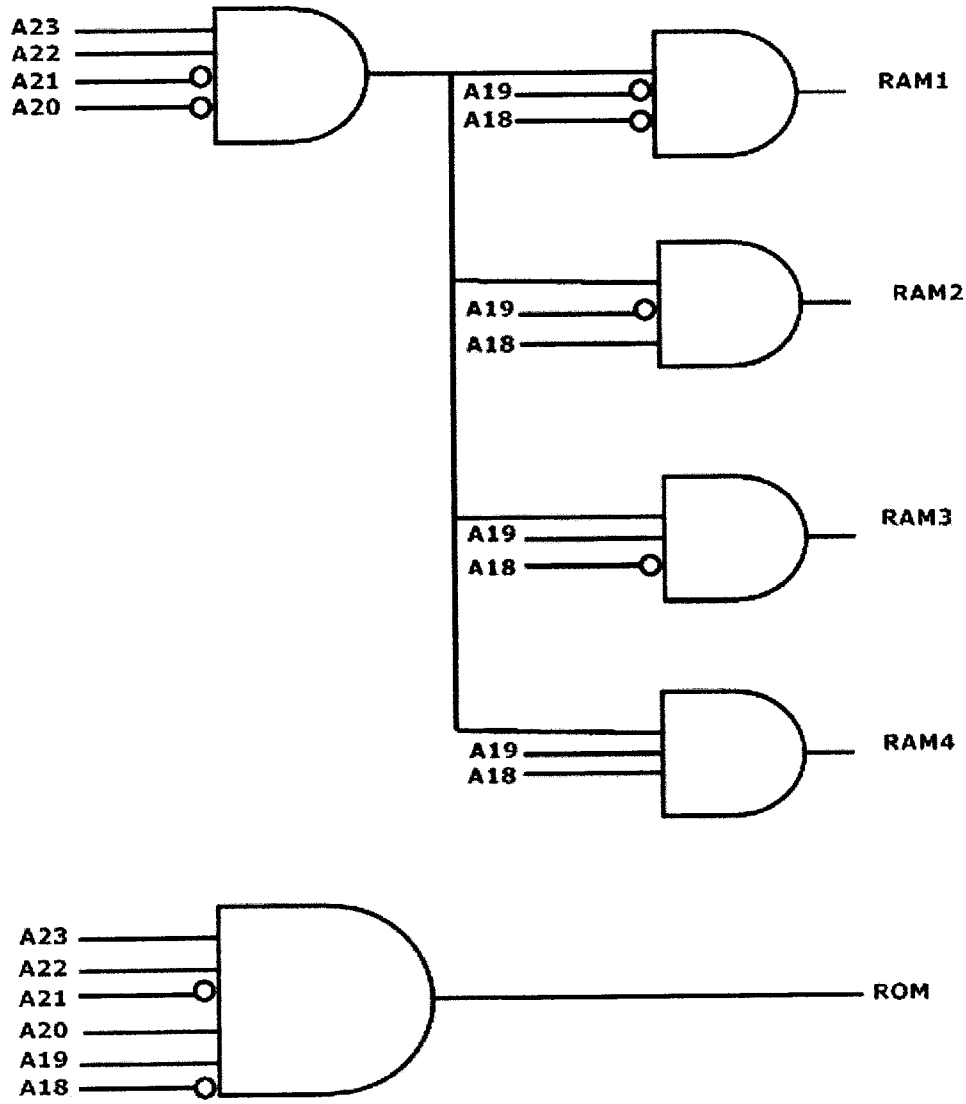


Figure Q5(c)

FINAL EXAMINATION

SEMESTER/SESSION : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT 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 : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT 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 $\overline{\textbf{TO}}$: Time-out bit
1 = After power-up, CLRWDT instruction, or SLEEP instruction
0 = A WDT time-out occurred

bit 3 $\overline{\textbf{PD}}$: 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 : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT 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 : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT 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 : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT 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	$\overline{TO}$	$\overline{PD}$	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/$\overline{DONE}$	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	$\overline{RBPU}$	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	$\overline{TO}$	$\overline{PD}$	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	—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$	---- --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 : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT 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/$\overline{borrow}$ 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

bit 0: **C:** Carry/$\overline{borrow}$ 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 $\overline{borrow}$ 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 : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT 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 : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT 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/$\overline{\text{DONE}}$	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/$\overline{\text{DONE}}$:** 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)

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 : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT 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 : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT 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	

FINAL EXAMINATION

SEMESTER/SESSION : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT CODE : BEE 3233

Data Transfer Instruction MC68000

Mnemonic	Meaning	Type	Operand Size	Operations
MOVE	Move	MOVE EAs, EAd	8, 16, 32	(EAs) → EAd
		MOVE EA,CCR	8	(EA) → CCR
		MOVE EA,SR	16	(EA) → SR
		MOVE SR, EA	16	SR → EA
		MOVE USP,An	32	USP → An
		MOVE An,USP	32	An → USP
		MOVEA EA,An	16, 32	(EA) → An
		MOVEQ #XXX,Dn	8	#XXX → Dn
MOVEM	Move multiple	MOVEM Reg_list,EA	16, 32	Reg_list → EA
		MOVEM EA,Reg_list	16, 32	(EA) → Reg_list
LEA	Load effective address	LEA EA,An	32	EA → An
	Exchange			
EXG	Swap	EXG Rx,Ry	32	Rx ↔ Ry
SWAP	Clear	SWAP Dn	16	Dn31:16 ↔ Dn15:0
CLR		CLR EA	8, 16, 32	0 → EA

FINAL EXAMINATION

SEMESTER/SESSION : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT CODE : BEE 3233

Compare and Test MC68000 Instruction

Mnemonic	Meaning	Type	Operand Size	Operation
CMP	Compare	CMP EA,Dn	8, 16, 32	N, Z, V, C
		CMPA EA,An	16, 32	N, Z, V, C
		CMPI #XXX,EA	8, 16, 32	N, Z, V, C
		CMPM (Ay)+,(AY)+	8, 16, 32	N, Z, V, C
TST	Test	TST EA	8, 16, 32	N, Z, V, C

Logical MC68000 Instruction

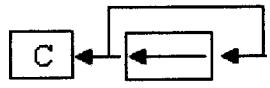
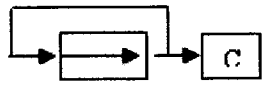
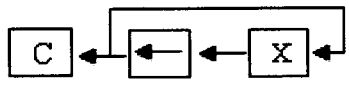
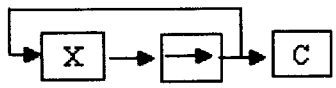
Mnemonic	Meaning	Type	Operand Size	Operation
AND	Logical AND	AND EA,Dn	8, 16, 32	(EA) · Dn → Dn
		AND Dn,EA	8, 16, 32	Dn · (EA) → EA
		ANDI #XXX,EA	8, 16, 32	#XXX · (EA) → EA
		ANDI #XXX,CCR	8	#XXX · CCR → CCR
		ANDI #XXX,SR	16	#XXX · SR → SR
OR	Logical OR	OR EA,Dn	8, 16, 32	(EA) + Dn → Dn
		OR Dn,EA	8, 16, 32	Dn + (EA) → EA
		ORI #XXX,EA	8, 16, 32	#XXX + (EA) → EA
		ORI #XXX,CCR	8	#XXX + CCR → CCR
		ORI #XXX,SR	16	#XXX + SR → SR
EOR	Logical exclusive-OR	EOR Dn,EA	8, 16, 32	Dn ⊕ (EA) → EA
		EORI #XXX,EA	8, 16, 32	#XXX ⊕ (EA) → EA
		EORI #XXX,CCR	8	#XXX ⊕ CCR → CCR
		EORI #XXX,SR	16	#XXX ⊕ SR → SR
NOT	Logical NOT	NOT EA	8, 16, 32	$(\overline{EA}) \rightarrow EA$

FINAL EXAMINATION

SEMESTER/SESSION : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT CODE : BEE 3233

Rotate MC68000 Instruction

Mnemonic	Meaning	Type	Operand Size	Operation
ROL	Rotate left	ROL #XXX,Dy ROL Dx,Dy ROL EA	8, 16, 32 8, 16, 32 8, 16, 32	
ROR	Rotate right	ROR #XXX,Dy ROR Dx,Dy ROR EA	8, 16, 32 8, 16, 32 8, 16, 32	
ROXL	Rotate left through extend	ROXL #XXX,Dy ROXL Dx,Dy ROXL EA	8, 16, 32 8, 16, 32 8, 16, 32	
ROXR	Rotate right through extend	ROXR #XXX,Dy ROXR Dx,Dy ROXR EA	8, 16, 32 8, 16, 32 8, 16, 32	

Bit Manipulation MC68000 Instruction

Mnemonic	Meaning	Type	Operand Size	Operation
BTST	Test a bit	BTST #XXX,EA BTST Dn,EA	8, 32 8, 32	 $\overline{EA}$ bit → Z
BSET	Test a bit and set	BSET #XXX,EA BSET Dn,EA	8, 32 8, 32	$\overline{EA}$ bit → Z 1 → EA bit
BCLR	Test a bit and clear	BCLR #XXX,EA BCLR Dn,EA	8, 32 8, 32	$\overline{EA}$ bit → Z 0 → EA bit
BCHG	Test a bit and change	BCHG #XXX,EA BCHG Dn,EA	8, 32 8, 32	$\overline{EA}$ bit → Z $\overline{EA}$ bit → EA bit

FINAL EXAMINATION

SEMESTER/SESSION : I/2010/2011
SUBJECT : MICROPROCESSOR AND MICROCONTROLLER

COURSE : 3 BEE
SUBJECT CODE : BEE 3233

Bcc Instruction

Instruction	Meaning	Arithmetic	If the test is true
BEQ	EQual to zero	U	Z=1
BNE	Not Equal to zero	U	Z=0
BMI	Minus	U	N=1
BPL	Plus	U	N=0
BCS/LO	Carry Set/LOwer	U	C=1
BCC/HS	Carry Clear/Higher or Same	U	C=0
BVS	oVerflow Set	S	V=1
BVC	oVerflow Clear	S	V=0
BGT	GreaTer than	S	Z+(N⊕V)=0
BLT	Less Than	S	N⊕V=1
BGE	Greater than or Equal	S	N⊕V=0
BLE	Less than or Equal	S	Z+(N⊕V)=0
BHI	Higher	U	C+Z=0
BLS	Lower than or Same	U	C+Z=1