Exam Question Paper

College/ Institute	CEDPS		
Department	ECE – TNE Program		
Exam Author(s)	Dr Zhengwen Huang		
Module Code	EE1655		
Module Title	Digital Systems and Microprocessors		
Month	June/August	Year	2019
Paper Type	Full		
Duration	3 Hours		
Question Instructions			
Are Calculators Permitted?	No		
Permitted Reference Materials	A Data sheet is attached to this exam paper.		
Required Stationery	Please use a separate answer book for each Section.		

SECTION A

1. Convert the following decimal number

   3275_{10} to

   a) Binary
   b) Hexadecimal
   c) 7,4,-2,-1 Binary Coded Decimal

   [5 marks]

2. In multiply parity error correction, 3 parity bits P_1, P_2, and P_3 are generated from 4 data bits $b_3b_2b_1b_0$. Consider the following definitions:

   - P_1 is the EVEN parity of $b_3b_2b_1$
   - P_2 is the EVEN parity of $b_3b_1b_0$
   - P_3 is the EVEN parity of $b_2b_1b_0$

   Determine the parity failures for single errors in each of the data bits b_3, b_2, $b_1$and b_0 and correct the error in the following data:

   $P_1P_2P_3b_3b_2b_1b_0$ = 0 1 0 1 0 0 1

   [5 marks]

3. Use the Truth Table Equivalence to determine whether the following equations are true or false.

   a) $\left(A + \overline{B}\right).\left(\overline{A} + \overline{B} + C\right) = A.C + A.\overline{B} + \overline{A}.\overline{B}$

   b) $\overline{\overline{(A \oplus B)} + (B \oplus C) + (A \oplus C)} = 1$

   [5 marks]

4. Obtain the MINIMAL 1st and 2nd Canonical forms of a logic circuit which indicates (F =1) when a 4-bit binary number (ABCD) is greater than 1 but less than 12

[5 marks]

5. The following equation is in its minimal NAND form. Obtain its MINIMAL 1st Canonical (AND/OR/NOT) form,

$$F = \overline{\overline{\overline{A}\,B}.\overline{B\overline{C}}.\overline{\overline{A}CD}.\overline{A\overline{B}\,\overline{D}}}$$

[5 marks]

6. A 4-bit shift register $Q_3Q_2Q_1Q_0$ has EXCLUSIVE-NOR feedback. The feedback function is

$$I_0 = \overline{Q_3 \oplus Q_2}$$

If the initial state is 1011, what sequence does the register generate when clocked?

[5 marks]

7. Design a sequential circuit to continuously output the sequence

0, 3, 4, 2, 5, 6, 7, 1, and repeat from 0

Include a detailed block diagram in your answer and obtain any necessary logic functions in their MINIMAL forms.

[5 marks]

8. Design a combinational logic system to convert 3-bit pure binary (b_2, b_1, b_0) into 3-bit Gray code (g_2, g_1, g_0). Using any cellular structure you can identify in your three equations, calculate the gray code of the following binary number

$$1110011110101_2$$

[5 marks]

9. Program the function of

$F = \sum (0, 1, 3, 7, 9, 12, 13, 15)$

Into

a) ONE 16 to 1 multiplexer.

b) 2 layer system of 4 to 1 multiplexer

Present your answers as a block diagram of the system with all the inputs and outputs appropriately labeled

[5 marks]

10. An engineer has designed the following circuit (FIGURE Q10). It works but takes up too much silicon area and consumes excessive power. Redesign the circuit and obtain the best possible design in terms of power consumption and size.

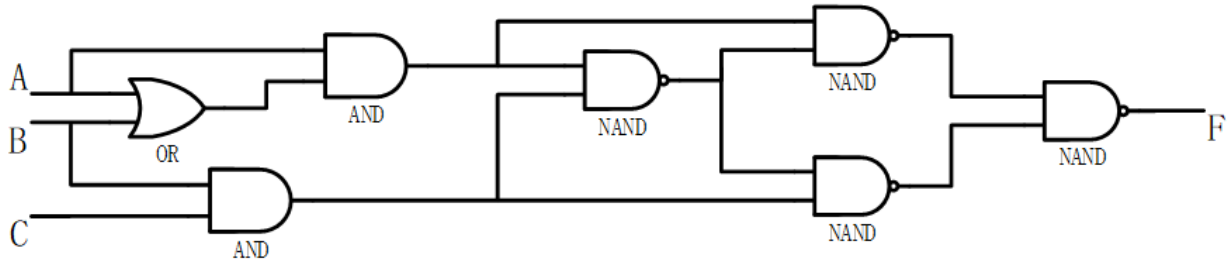


FIGURE Q10.

[5 marks]

Section B

11. Read each of the following statements about the PIC16F877A microcontroller and identify whether it is true or false and explain the reason.

a) The program counter has 13 bits.
b) The program counter has a stack with 10 levels.
c) It allows more data locations by grouping the data into 6 separate banks.
d) It has 35 instructions.
e) STATUS register only appears in the first bank.
f) The size of program memory is 14k Bytes.
g) The width of data memory is 10bits.
h) The chip has 40 pins.
i) It is Von Neuman architecture.
j) It is a reduced instruction set computer (RISC).

[10 marks]

12. Provide the hexadecimal and binary representation of the following value in the data memory (8-bits registers) of PIC microcontroller.

a) 100
b) -10
c) 127
d) 36
e) -98

[5 marks]

13. a) The following assembly language code is written for a PIC16F877A chip:

```
MOVF  PORTA
ADDWF PORTB
MOVWF PORTC
ANDLW   255
```

Write down the 14-bit binary machine code for each row of the code and explain the meaning of each instruction. (instruction table is attached at the end of exam paper)

[8 marks]

b) The following assembly language code is used to change data memory bank

```
BSF     STATUS,RP0
BCF     STATUS, RP1
```

After the execution of the above code, which bank is selected? RP1 and RP0 are defined with the following FIGURE Q13.

RP1	RP0	
0	0	memory bank 0
0	1	memory bank 1
1	0	memory bank 2
1	1	memory bank 3

FIGURE Q13.

[2 marks]

14. The Port B of a PIC microcontroller is connected with an 8-segment display as shown in Figure Q15 (a).

a) Write down all the value on PORT B of the PIC microcontroller for showing the letters "A", "b", "C", "d", "E", "F", "G" and "H" in in FIGURE Q15(b).

Continued on the next page

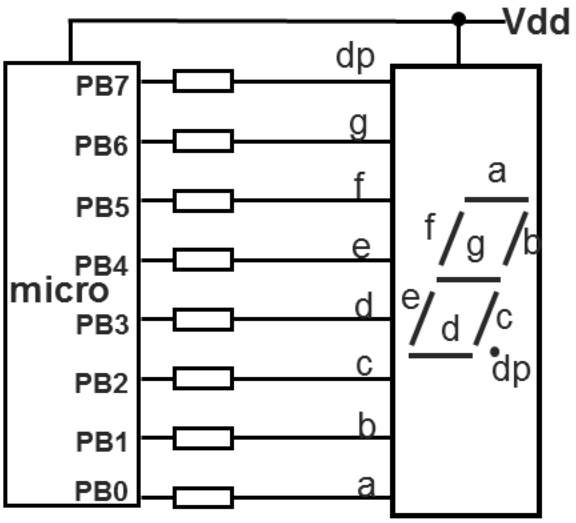


FIGURE Q15 (a).

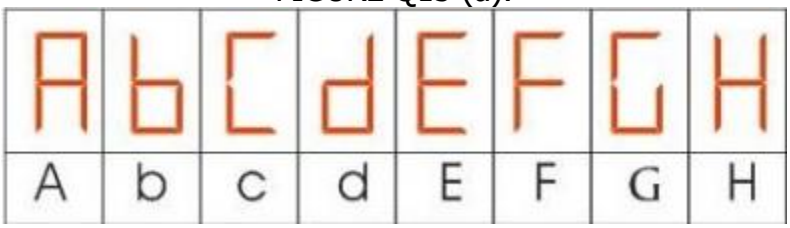


FIGURE Q15 (b).

[4 marks]

b) Write a piece of assembly code that can show the letters one by one on the 8-segment display.

[6 marks]

15. Below is a segment of code written for a PIC16F877A chip in assembly language.

```
DELAY   MOVLW    0x1F
        MOVWF    COUNT1
        MOVLW    0x0A
        MOVWF    COUNT2
DELAY1  DECFSZ   COUNT1,F
        GOTO     DELAY2
        GOTO     COMPLETE
DELAY2  DECFSZ   COUNT2,F
        GOTO     DELAY2
        GOTO     DELAY1
COMPLETE RETURN
```

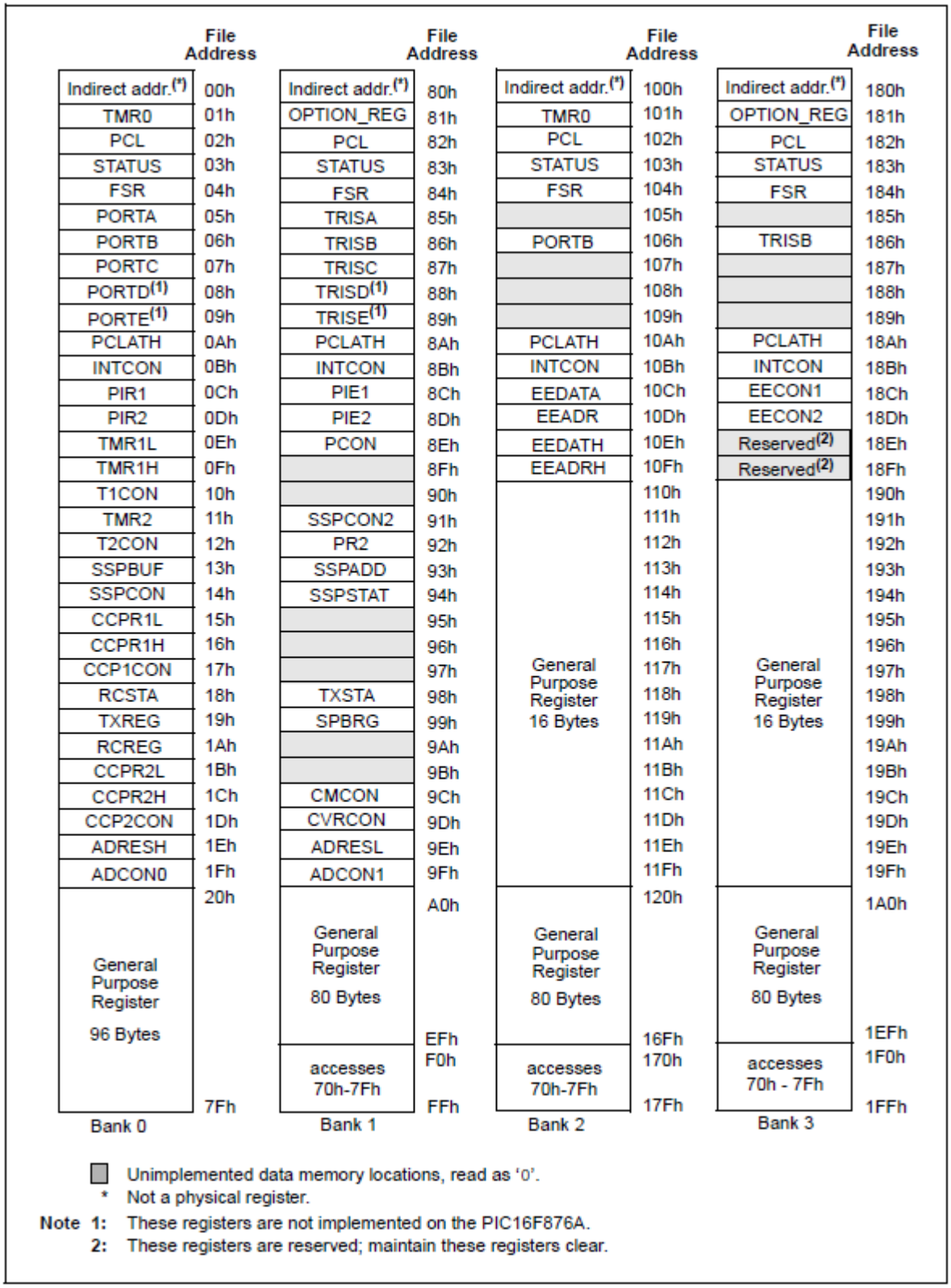
Continued on the next page

a) Calculate the number of instructions to run when DELAY is called.

[5 marks]

b) Assume that each PIC instruction takes 1 μs to execute. Calculate the total execution time to run this segment of code (Instruction table is attached at the end of exam paper). Note: Each of instructions of CALL, RETURN and GOTO takes 2 μs to execute.

[5 marks]


File Address
Indirect addr.(*) 00h
TMR0 01h
PCL 02h
STATUS 03h
FSR 04h
PORTA 05h
PORTB 06h
PORTC 07h
PORTD(1) 08h
PORTE(1) 09h
PCLATH 0Ah
INTCON 0Bh
PIR1 0Ch
PIR2 0Dh
TMR1L 0Eh
TMR1H 0Fh
T1CON 10h
TMR2 11h
T2CON 12h
SSPBUF 13h
SSPCON 14h
CCPR1L 15h
CCPR1H 16h
CCP1CON 17h
RCSTA 18h
TXREG 19h
RCREG 1Ah
CCPR2L 1Bh
CCPR2H 1Ch
CCP2CON 1Dh
ADRESH 1Eh
ADCON0 1Fh
20h
General Purpose Register
96 Bytes
7Fh
Bank 0
File Address
Indirect addr.(*) 80h
OPTION_REG 81h
PCL 82h
STATUS 83h
FSR 84h
TRISA 85h
TRISB 86h
TRISC 87h
TRISD(1) 88h
TRISE(1) 89h
PCLATH 8Ah
INTCON 8Bh
PIE1 8Ch
PIE2 8Dh
PCON 8Eh
8Fh
90h
SSPCON2 91h
PR2 92h
SSPADD 93h
SSPSTAT 94h
95h
96h
97h
TXSTA 98h
SPBRG 99h
9Ah
9Bh
CMCON 9Ch
CVRCON 9Dh
ADRESL 9Eh
ADCON1 9Fh
A0h
General Purpose Register
80 Bytes
EFh
F0h
accesses 70h-7Fh
FFh
Bank 1
File Address
Indirect addr.(*) 100h
TMR0 101h
PCL 102h
STATUS 103h
FSR 104h
105h
PORTB 106h
107h
108h
109h
PCLATH 10Ah
INTCON 10Bh
EEDATA 10Ch
EEADR 10Dh
EEDATH 10Eh
EEADRH 10Fh
110h
111h
112h
113h
114h
115h
116h
117h
118h
119h
11Ah
11Bh
11Ch
11Dh
11Eh
11Fh
120h
General Purpose Register
80 Bytes
16Fh
170h
accesses 70h-7Fh
17Fh
Bank 2
File Address
Indirect addr.(*) 180h
OPTION_REG 181h
PCL 182h
STATUS 183h
FSR 184h
185h
TRISB 186h
187h
188h
189h
PCLATH 18Ah
INTCON 18Bh
EECON1 18Ch
EECON2 18Dh
Reserved(2) 18Eh
Reserved(2) 18Fh
190h
191h
192h
193h
194h
195h
196h
197h
198h
199h
19Ah
19Bh
19Ch
19Dh
19Eh
19Fh
1A0h
General Purpose Register
80 Bytes
1EFh
1F0h
accesses 70h - 7Fh
1FFh
Bank 3
General Purpose Register 16 Bytes
Unimplemented data memory locations, read as '0'.
* Not a physical register.
Note 1: These registers are not implemented on the PIC16F876A.
2: These registers are reserved; maintain these registers clear.

Mnemonic, Operands		Description	Cycles	14-Bit Opcode				Status Affected	Notes
				MSb			LSb		
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.