POLITEKNIK
MALAYSIA

BAHAGIAN PEPERIKSAAN DAN PENILAIAN
JABATAN PENDIDIKAN POLITEKNIK
KEMENTERIAN PENDIDIKAN TINGGI

JABATAN KEJURUTERAAN ELEKTRIK

PEPERIKSAAN AKHIR

SESI DISEMBER 2015

DEC5052: EMBEDDED SYSTEM APPLICATIONS

TARIKH : 11 APRIL 2016
MASA : 8.30 AM – 10.30 AM (2 JAM)

Kertas ini mengandungi **EMPAT BELAS (14)** halaman bercetak.
Bahagian A: Struktur (4 soalan)
Bahagian B: Esei (2 soalan)
Dokumen sokongan yang disertakan : Tiada

JANGAN BUKA KERTAS SOALAN INI SEHINGGA DIARAHKAN

(CLO yang tertera hanya sebagai rujukan)

SECTION A: 40 MARKS

BAHAGIAN A: 40 MARKAH

INSTRUCTION:

This section consists of **FOUR (4)** structured questions. Answer **ALL** questions.

ARAHAN:

*Bahagian ini mengandungi **EMPAT (4)** soalan berstruktur. Jawab **SEMUA** soalan.*

QUESTION 1

SOALAN 1

CLO1 C1

(a) A microcontroller a heart of embedded system. Describe this statement by referring to the definition of embedded system.
Mikropengawal digambarkan sebagai jantung bagi sistem terbenam. Jelaskan kenyataan ini dengan merujuk kepada definisi sistem terbenam.

[3 marks]

[3 markah]

CLO1 C3

(b) A programmer decides to use two DC motors at pin RB6 and RB7. While, the LED at pin RA5 and two digital sensors at pin RB0 and RB1 in his circuit. Write the input output initialization for the program using bit addressable format in C language.
Seorang pengaturcara memutuskan untuk menggunakan dua motor AT pada pin RB6 dan RB7. Manakala LED pada pin RA5 dan dua pengesan digital pada pin RB0 dan RB1 dalam litar beliau. Tuliskan inisialisasi masukan dan keluaran menggunakan format bit pengalamatan dalam Bahasa C.

[6 marks]

[6 markah]

CLO2 C3

(c) Build a C program for main() function to make LED1 on port RB0 and LED2 on port RB1 blink alternately for each 2 second continuously. You are given a sub function for time delay in Figure 1 (c).

Bina satu atucara dalam program C untuk fungsi main() bagi mengawal LED1 pada port RB0 dan LED2 pada port RB1 supaya berkelip setiap 2 saat secara selang seli berterusan. Anda diberi sub fungsi bagi 'delay' seperti Rajah 1 (c):

```
void Delay_1ms(unsigned int x)
{
        for(;x>0;x--)
        __delay_ms(1);
}
```

Figure 1 (c) / *Rajah 1 (c)*

[6 marks]
[*6 markah*]

QUESTION 2
SOALAN 2

CLO1 C2 (a) Explain the function of TxCON (Timer Control) in PIC to operate in 16-bit and 8-bit modes.

Terangkan fungsi TxCON dalam PIC untuk dikendalikan dalam mod 16-bit dan 8-bit.

[3 marks]
[*3 markah*]

CLO1 C3 (b) 1-Hz external clock is being fed into pin T0CKI (RA4) and PORTB is connected to 8 LEDs as illustrate in Figure 2(c). Develop a C program for Counter 0 in 8-bit mode to count up and send data of the TMR0L count to PORTB. Start the count at 0H.

Frekuensi 1-Hz dari jam luaran dipacu pada pin T0CKI (RA4) dan lapan LED disambungkan ke PORTB seperti ditunjukkan dalam Rajah 2(c). Binakan aturcara dalam Bahasa C untuk Pembilang 0 dalam mod 8-bit untuk membilang secara menaik dan hantar data pada TMR0L pada PORTB. Mulakan kiraan pada nilai 0H.

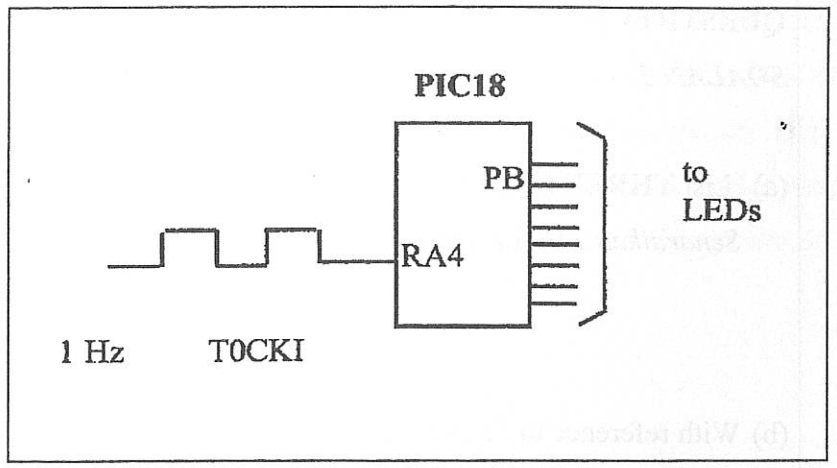

Figure 2 (b): Connection of clock and LED using PIC18
Rajah 2 (b): Sambungan jam dan LED menggunakan PIC18

[6 marks]
[*6 markah*]

CLO2 C3 (c) The Timer0 in PIC18F is used to get 13ms delay. Show the steps to get the value of T0CON based on information in Figure 2 (c).

Pemasa Timer0 dalam PIC18F digunakan untuk menghasilkan lengah masa selama 13ms. Tunjukkan langkah-langkah untuk mendapatkan nilai T0CON berdasarkan maklumat dalam Rajah 2 (c).

TMR0H = 0x02
TMR0L = 0x12
F_{OSC} = 20MHz

Figure 2 (c) / *Rajah 2 (c)*

[6 marks]
[*6 markah*]

QUESTION 3

SOALAN 3

CLO1 C1

(a) List **THREE** (3) sources of PIC 18 interrupt signal.

*Senaraikan **TIGA** (3) punca sampukan pada PIC18.*

[3 marks]

[3 markah]

CLO1 C2

(b) With reference to Figure 3 (b), produce C program that used to enable external hardware INT0 (RB0) to push interrupt vector into interrupt Service Routine (ISR). Use Appendix A1 to A3 as references.

Dengan merujuk kepada Rajah 3 (b), Berikan contoh program C yang digunakan bagi membolehkan sampukan luaran INT0 (RB0) untuk menolak "Interrupt Vector" ke dalam "Interrupt Service Routine" (ISR). Gunakan Lampiran A1 ke A3 sebagai panduan.

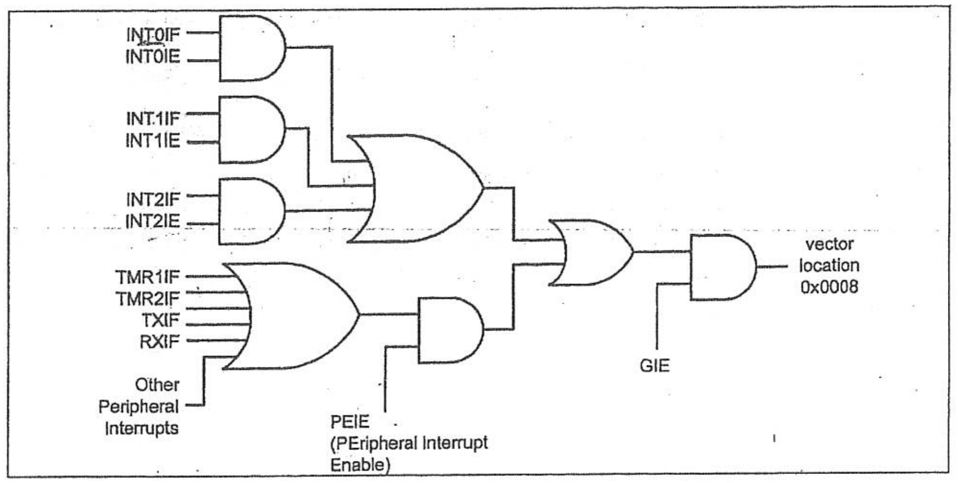

Figure 3 (b) / *Rajah 3 (b)*

[5 marks]

[5 markah]

CLO1 C3

(c) Referring to Figure 3 (c), build a C program with input on Pin RB1 (INT1) is connected to a pulse generator and output on pin RB7 is connected to the LED. This program will toggle the LED on the falling edge of the pulse. LED is turned on and off at the same rate as the pulses applied to the INT1 pin. Use Appendix A1 to A3 as references.

Merujuk kepada Rajah 3 (c), bina atucara C dengan input pada Pin RB1 (INT1) disambungkan kepada "pulse generator" dan output pada pin RB7 disambungkan ke LED. Program ini akan menyebabkan LED berkelip di "falling edge of the pulse ". Dengan kata lain, LED dihidupkan dan di luar pada kadar yang sama seperti denyutan digunakan untuk pin INT1 itu. Gunakan Lampiran A1 ke A3 sebagai panduan.

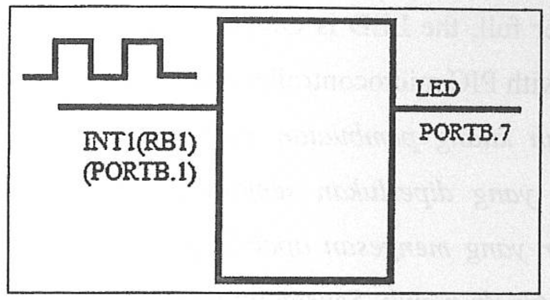

Figure 3 (c) / *Rajah 3 (c)*

[7 marks]

[7 markah]

QUESTION 4

SOALAN 4

CLO1 C2

(a) Explain the Analogue-to-Digital Converter (ADC) module in the PIC.

Terangkan berkenaan modul Analogue-to-Digital (ADC) di dalam PIC.

[3 marks]

[3 markah]

CLO1 C3

(b) An ADC module inside PIC microcontroller used to convert analogue signal from temperature sensor LM35/LM34. If the V_{REF} (+) of ADC module is taken from external source. Sketch a simple circuit to interface temperature sensor LM35/LM34 with PIC microcontroller.

Modul ADC di dalam pengawal mikro PIC telah digunakan untuk menukarkan isyarat analog dari penderia suhu LM35/LM34. Jika V_{REF} (+) bagi modul ADC ini diambil daripada sumber luar. Lakarkan dengan ringkas litar antaramuka di antara penderia suhu LM35/LM34 dengan pengawal mikro PIC.

[5 marks]

[5 markah]

CLO2
C5

(c) A manufacturing plant uses two tanks to store certain liquid chemical that are required in a manufacturing process. Each tank has a sensor that detects when the chemical level drop 25% from full tank level. The sensor produces a HIGH level of 5V when the tank is more than one-quarter full. When the volume of chemical in a tank drops to one-quarter full, the sensor puts out a LOW level of 0V. It is required a red LED display come on when at least one of the tanks falls to the quarter full level. Refer to Figure 4 (c), if tank A or tank B or both are below one-quarter full, the LED is ON, build the embedded system to replace negative-OR gate with PIC microcontroller and write C program to control the operation.

Sebuah kilang pembuatan menggunakan dua tangki untuk menyimpan cecair kimia yang diperlukan semasa proses pembuatan. Setiap tangki mempunyai sensor yang mengesan apabila paras cecair kimia turunkan 25% dari bacaan pada tangki penuh. Sensor ini menghasilkan logik HIGH iaitu 5V apabila tangki adalah lebih daripada satu perempat penuh. Apabila jumlah cecair kimia dalam tangki surut kepada satu perempat penuh, sensor akan mengeluarkan logic LOW iaitu 0V. Ia memerlukan LED merah untuk menyala apabila sekurang-kurangnya salah satu daripada tangki jatuh ke tahap suku penuh. Dengan merujuk Rajah 4 (c), binakan satu sistem terbenam untuk menggantikan negatif-OR gate dengan PIC mikropengawal dengan menulis aturcara C untuk mengawal operasi iaitu jika tangki A atau tangki B atau kedua-duanya berada di bawah satu perempat penuh, maka LED akan dihidupkan.

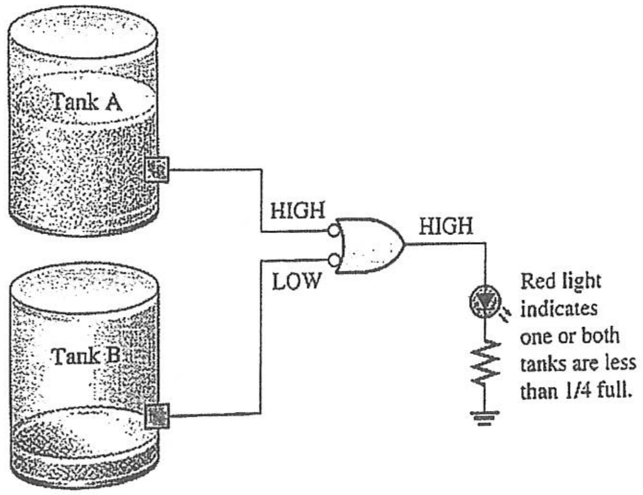
Tank A
Tank B
HIGH
LOW
HIGH
Red light indicates one or both tanks are less than 1/4 full.

Figure 4 (c) / *Rajah 4 (c)*

[7 marks]
[*7 markah*]

SECTION B: 40 MARKS
BAHAGIAN B: 40 MARKAH

INSTRUCTION:
This section consists of **TWO (2)** essay questions. Answer **ALL** questions.

ARAHAN:
*Bahagian ini mengandungi **DUA (2)** soalan esei. Jawab **SEMUA** soalan.*

QUESTION 1
SOALAN 1

CLO2
C3

Refer to Figure B1, construct a C program which uses external interrupt feature in PIC microcontroller to perform the following task:
"*Every time INT0 is activated, it toggles the LED while at the same time data is being transferred from PORTC to PORTD.*"
Use Appendix A1 to A3 as a reference.
Dengan merujuk kepada Rajah B1, binakan aturcara C yang menggunakan sampukan luaran untuk mikropengawal PIC untuk melakukan tugas berikut:
"Setiap kali INT0 diaktifkan, ia akan togol LED dan dalam waktu yang sama data dipindahkan dari PORTC ke PORTD."
Gunakan Lampiran A1 hingga A3 sebagai rujukan.

[20 marks]
[20 *markah*]

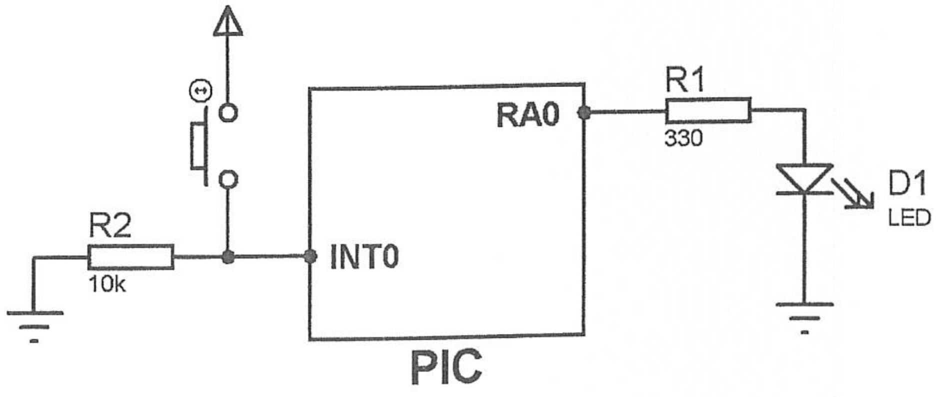

Figure B1/*Rajah B1*

QUESTION 2

SOALAN 2

CLO2 C5 Design an embedded system complete with programming in C, which can be used to demonstrate a LED dimmer using internal PWM in PIC microcontroller. Schematic sketching must include a LED that is connected to the output of a CCP pin through a resistor to limit the current and two Push Button that act as switches to adjust the PWM duty cycle. The first switch use as increment and the second switch use as decrement the duty cycle. Set PWM period to 1 kHz with external OSC at 20MHz and prescale is 1:32. Use Appendix A4 to A5 as a reference.

Reka sebuah sistem terbenam bersama dengan program C, yang boleh mengawal kecerahan LED dengan menggunakan PWM yang terdapat dalam cip PIC. Lakaran litar yang direkabentuk mestilah terdiri daripada satu LED yang disambungkan pada pin CCP melalui perintang untuk menghadkan aliran arus pada output dan dua suis yang bertindak sebagai pengubah nilai kitar tugas PWM. Suis pertama sebagai peningkat dan suis kedua sebagai penurun nilai kitar tugas. Setkan PWM kepada 1kHz dengan OSC luar yang digunakan adalah 20MHz dan pre—skala sebagai 1:32. Gunakan Lampiran A4 hingga A5 sebagai rujukan.

PWM period = (PR2 +1) x 4 x tOSC x (TMR2 prescale).

Duty Cycle period = (CCPR1L:CCP1CON<5:4>) x tOSC x (TMR2 prescale).

[20 marks]

[20 *markah*]

SOALAN TAMAT

APPENDIX

Appendix A1

REGISTER 9-1: INTCON: INTERRUPT CONTROL REGISTER

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/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF$^{(1)}$
bit 7							bit 0

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

bit 7 **GIE/GIEH:** Global Interrupt Enable bit
When IPEN = 0:
1 = Enables all unmasked interrupts
0 = Disables all interrupts
When IPEN = 1:
1 = Enables all high priority interrupts
0 = Disables all high priority interrupts

bit 6 **PEIE/GIEL:** Peripheral Interrupt Enable bit
When IPEN = 0:
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts
When IPEN = 1:
1 = Enables all low priority peripheral interrupts
0 = Disables all low priority peripheral interrupts

bit 5 **TMR0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 overflow interrupt
0 = Disables the TMR0 overflow interrupt

bit 4 **INT0IE:** INT0 External Interrupt Enable bit
1 = Enables the INT0 external interrupt
0 = Disables the INT0 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 **TMR0IF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1 **INT0IF:** INT0 External Interrupt Flag bit
1 = The INT0 external interrupt occurred (must be cleared in software)
0 = The INT0 external interrupt did not occur

bit 0 **RBIF:** RB Port Change Interrupt Flag bit$^{(1)}$
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

APPENDIX

Appendix A2

REGISTER 9-2: INTCON2: INTERRUPT CONTROL REGISTER 2

R/W-1	R/W-1	R/W-1	R/W-1	U-0	R/W-1	U-0	R/W-1
$\overline{\text{RBPU}}$	INTEDG0	INTEDG1	INTEDG2	—	TMR0IP	—	RBIP
bit 7							bit 0

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

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

bit 6 **INTEDG0:** External Interrupt 0 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 5 **INTEDG1:** External Interrupt 1 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 4 **INTEDG2:** External Interrupt 2 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 3 **Unimplemented:** Read as '0'

bit 2 **TMR0IP:** TMR0 Overflow Interrupt Priority bit
1 = High priority
0 = Low priority

bit 1 **Unimplemented:** Read as '0'

bit 0 **RBIP:** RB Port Change Interrupt Priority bit
1 = High priority
0 = Low priority

APPENDIX

Appendix A3

REGISTER 9-3: INTCON3: INTERRUPT CONTROL REGISTER 3

R/W-1	R/W-1	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
INT2IP	INT1IP	—	INT2IE	INT1IE	—	INT2IF	INT1IF
bit 7							bit 0

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

bit 7 **INT2IP:** INT2 External Interrupt Priority bit
1 = High priority
0 = Low priority

bit 6 **INT1IP:** INT1 External Interrupt Priority bit
1 = High priority
0 = Low priority

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

bit 4 **INT2IE:** INT2 External Interrupt Enable bit
1 = Enables the INT2 external interrupt
0 = Disables the INT2 external interrupt

bit 3 **INT1IE:** INT1 External Interrupt Enable bit
1 = Enables the INT1 external interrupt
0 = Disables the INT1 external interrupt

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

bit 1 **INT2IF:** INT2 External Interrupt Flag bit
1 = The INT2 external interrupt occurred (must be cleared in software)
0 = The INT2 external interrupt did not occur

bit 0 **INT1IF:** INT1 External Interrupt Flag bit
1 = The INT1 external interrupt occurred (must be cleared in software)
0 = The INT1 external interrupt did not occur

APPENDIX

Appendix A4

REGISTER 16-1: CCP1CON REGISTER (ECCP1 MODULE, 40/44-PIN DEVICES)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
bit 7							bit 0

bit 7-6 **P1M1:P1M0**: Enhanced PWM Output Configuration bits

<u>If CCP1M3:CCP1M2 = 00, 01, 10:</u>
xx = P1A assigned as Capture/Compare input/output; P1B, P1C, P1D assigned as port pins
<u>If CCP1M3:CCP1M2 = 11:</u>
00 = Single output: P1A modulated; P1B, P1C, P1D assigned as port pins
01 = Full-bridge output forward: P1D modulated; P1A active; P1B, P1C inactive
10 = Half-bridge output: P1A, P1B modulated with dead-band control; P1C, P1D assigned as port pins
11 = Full-bridge output reverse: P1B modulated; P1C active; P1A, P1D inactive

bit 5-4 **DC1B1:DC1B0**: PWM Duty Cycle bit 1 and bit 0

<u>Capture mode:</u>
Unused.
<u>Compare mode:</u>
Unused.
<u>PWM mode:</u>
These bits are the two LSbs of the 10-bit PWM duty cycle. The eight MSbs of the duty cycle are found in CCPR1L.

bit 3-0 **CCP1M3:CCP1M0**: Enhanced CCP Mode Select bits

0000 = Capture/Compare/PWM off (resets ECCP module)
00.01 = Reserved
0010 = Compare mode, toggle output on match
0011 = Capture mode
0100 = Capture mode, every falling edge
0101 = Capture mode, every rising edge
0110 = Capture mode, every 4th rising edge
0111 = Capture mode, every 16th rising edge
1000 = Compare mode, initialize CCP1 pin low, set output on compare match (set CCP1IF)
1001 = Compare mode, initialize CCP1 pin high, clear output on compare match (set CCP1IF)
1010 = Compare mode, generate software interrupt only, CCP1 pin reverts to I/O state
1011 = Compare mode, trigger special event (ECCP resets TMR1 or TMR3, sets CC1IF bit)
1100 = PWM mode; P1A, P1C active-high; P1B, P1D active-high
1101 = PWM mode; P1A, P1C active-high; P1B, P1D active-low
1110 = PWM mode; P1A, P1C active-low; P1B, P1D active-high
1111 = PWM mode; P1A, P1C active-low; P1B, P1D active-low

APPENDIX

Appendix A5

REGISTER 13-1: T2CON: TIMER2 CONTROL REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit 7							bit 0

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

bit 6-3 **T2OUTPS3:T2OUTPS0**: Timer2 Output Postscale Select bits

0000 = 1:1 Postscale
0001 = 1:2 Postscale
•
•
•
1111 = 1:16 Postscale

bit 2 **TMR2ON**: Timer2 On bit

1 = Timer2 is on
0 = Timer2 is off

bit 1-0 **T2CKPS1:T2CKPS0**: Timer2 Clock Prescale Select bits

00 = Prescaler is 1
01 = Prescaler is 4
1x = Prescaler is 16