

SULIT



**KEMENTERIAN PENDIDIKAN TINGGI
JABATAN PENDIDIKAN POLITEKNIK DAN KOLEJ KOMUNITI**

**BAHAGIAN PEPERIKSAAN DAN PENILAIAN
JABATAN PENDIDIKAN POLITEKNIK DAN KOLEJ KOMUNITI
KEMENTERIAN PENDIDIKAN TINGGI**

JABATAN KEJURUTERAAN ELEKTRIK

PEPERIKSAAN AKHIR

SESI I : 2025/2026

DEC40053: EMBEDDED SYSTEM APPLICATIONS

TARIKH : 23 NOVEMBER 2025

MASA : 8.30 PAGI – 10.30 PAGI (2 JAM)

Kertas soalan ini mengandungi **LAPAN (8)** halaman bercetak.

Bahagian A: Subjektif (3 soalan)

Bahagian B: Esei (2 soalan)

Dokumen sokongan yang disertakan : *PIC18 Data Sheet*

JANGAN BUKA KERTAS SOALAN INI SEHINGGA DIARAHKAN

(CLO yang tertera hanya sebagai rujukan)

SULIT

SECTION A : 60 MARKS**BAHAGIAN A : 60 MARKAH****INSTRUCTION:**

This section consists of **THREE (3)** structured questions. Answer **ALL** questions.

ARAHAN:

Bahagian ini mengandungi TIGA (3) soalan berstruktur. Jawab semua soalan

QUESTION 1**SOALAN 1**

- CLO1 (a) Explain the function of PORTx register in I/O with an example of C language program using bit addressable.

Terangkan fungsi pendaftar PORTx di dalam I/O berserta contoh aturcara Bahasa C menggunakan format pengalamanan bit.

[4 marks]

[4 markah]

- CLO1 (b) Figure A1 (b) shows the connection between input devices and output devices with PIC microcontroller. Write the C language to configure the input/output port using bit addressable and byte addressable.

Rajah A1(b) menunjukkan sambungan peranti masukan dan peranti keluaran dengan pengawal mikro PIC. Tulis aturcara Bahasa C untuk mengkonfigurasi pin masukan/keluaran menggunakan format pengalamanan bit dan byte.

[8 marks]

[8 markah]

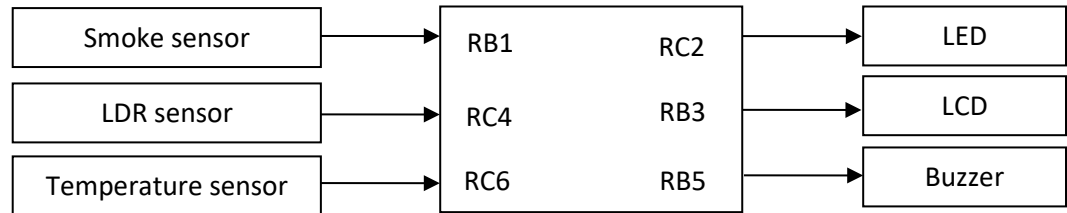


Figure A1(b)/ Rajah A1(b)

- CLO1 (c) Given a crystal oscillator frequency = 20MHz and the value of TMR0H:TMR0L = D5A8 H. Referring to Appendix 1, calculate the time delay generated by Timer0 if T0CON is set to 0x05.

Diberi frekuensi pengayun = 20MHz dan nilai TMR0H:TMR0L = D5A8H. Merujuk kepada Lampiran 1, kirakan lengah masa yang dijana oleh Timer0 jika T0CON disetkan kepada 0x05.

[8 marks]

[8 markah]

QUESTION 2

SOALAN 2

- CLO1 (a) Explain the function of TMRxH and TMRxL registers in Timer 0 for 8 bit and 16 bit counting.

Terangkan fungsi pendaftar TMRxH dan TMRxL dalam Pemasa 0 untuk pengiraan 8 bit dan 16 bit

[4 marks]

[4 markah]

- CLO1 (b) Write C instructions for PIC18 to initialize INT1 external hardware interrupt and its Interrupt Service Routine (ISR).

Tulis arahan C untuk PIC18 memulakan sampukan perkakasan luaran INT1 dan Interrupt Service Routine (ISR) miliknya.

[8 marks]

[8 markah]

- CLO1 (c) Write C instructions for PIC18 to initialize TIMER0 counter interrupt and its Interrupt Service Routine (ISR).

Tulis arahan C untuk PIC18 memulakan sampukan pembilang TIMER0 dan Interrupt Service Routine (ISR) miliknya.

[8 marks]

[8 markah]

QUESTION 3

SOALAN 3

- CLO1 (a) Compare **TWO (2)** characteristics of the interrupt and polling methods in a PIC18F4550 microcontroller.

Bezakan DUA (2) ciri kaedah sampukan dan tinjauan dalam mikropengawal PIC18F4550.

[4 marks]

[4 markah]

CLO1

- (b) ADC (analog-to-digital converter) and PWM (pulse-width modulation) are both methods of controlling and manipulating analog signals using digital devices, but they differ in their approaches and applications. Compare **THREE (3)** differences between ADC and PWM.

*ADC (penukar analog-ke-digital) dan PWM (modulasi lebar denyut) ialah kedua-dua kaedah mengawal dan memanipulasi isyarat analog menggunakan peranti digital, tetapi ia berbeza dalam pendekatan dan aplikasinya. Bandingkan **TIGA (3)** perbezaan antara ADC dan PWM.*

[6 marks]

[6 markah]

CLO1

- (c) In figure A3(c), a 10-bit potentiometer is used to light up 10 LEDs as an output of ADC Experiment. Given the ADC voltage reference, $V_{ref} = 5V$. Calculate the output of LED in hexadecimal value when the potentiometer is set at 50% and 80%.

Dalam rajah A3(c), 10bit potentiometer digunakan untuk menghidupkan 10 unit LED sebagai keluaran Eksperimen ADC. Diberi nilai $V_{ref} = 5V$, kirakan keluaran LED dalam nilai hexadecimal apabila potentiometer disetkan pada 50% dan 80%.

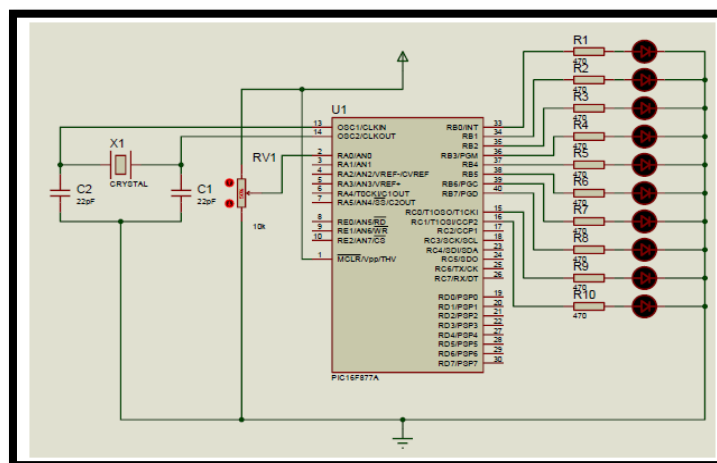


Figure A3(c)/ Rajah A3(c)

[10 marks]

[10 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***

CLO1 An assistant engineer has been assigned to design a speed limit display system using an LCD display that can be adjusted based on weather conditions to enhance road safety during adverse weather, such as heavy rain. The system should display a standard speed limit of 90 km/h under normal conditions. When heavy rain is detected, the LCD display should automatically switch to a reduced speed limit of 60 km/h to improve safety. The project incorporates an interrupt mechanism to enhance the system's responsiveness. Based on the given Table B1, determine the program in C language by using PIC18 complete with schematic circuit for the speed limit display system.

Table B1: Weather – Speed limit Reference

| Weather state | Rain sensor | Displayed speed |
|---------------|-------------|-----------------|
| Normal (dry) | HIGH | 90 km/h |
| Heavy rain | LOW | 60 km/h |

Seorang pembantu jurutera telah ditugaskan untuk merekabentuk sistem isyarat LCD had kelajuan yang boleh disesuaikan berdasarkan keadaan cuaca untuk meningkatkan keselamatan jalan raya semasa cuaca buruk, seperti hujan lebat. Sistem ini harus memaparkan had kelajuan standard sebanyak 90 km/j dalam keadaan normal. Apabila hujan lebat dikesan, paparan LCD harus secara automatik beralih kepada had kelajuan yang dikurangkan sebanyak 60 km/j untuk meningkatkan keselamatan. Projek ini menggabungkan mekanisme gangguan untuk meningkatkan responsiviti sistem. Berdasarkan Jadual B1 yang diberikan, tentukan pengaturcaraan dalam bahasa C

dengan menggunakan PIC18, lengkap dengan litar skematik untuk sistem paparan had kelajuan.

[20 marks]

[20 markah]

QUESTION 2

SOALAN 2

CLO2

A parking garage with an automatic lighting control system is developed to improve energy efficiency. The garage is divided into multiple sections, and each section is designed to have lighting controlled by pulse-width modulation (PWM) to adjust brightness levels based on occupancy. The system uses a PIR motion sensor to detect the presence of vehicles or people in a section, adjusting the brightness of the LED lights accordingly. The PIR sensor is connected to PORT B of a PIC18F4550 microcontroller. The LED lights are connected to PORT C and are dimmed using PWM. Develop the program in C language using the PIC18F4550 microcontroller to control the brightness of the lights based on the motion detected by the PIR sensor with reference to Table B2. In your design, include the schematic circuit for the parking garage automatic lighting control system.

Sebuah garaj kereta dengan sistem kawalan lampu automatik dibangunkan untuk meningkatkan kecekapan tenaga. Garaj ini dibahagikan kepada beberapa bahagian, dan setiap bahagian direka untuk mempunyai pencahayaan yang dikawal oleh modulasi lebar denyut (PWM) bagi melaras tahap kecerahan berdasarkan kehadiran. Sistem ini menggunakan pengesan gerakan PIR untuk mengesan kehadiran kenderaan atau orang di dalam sesuatu bahagian, dan melaraskan kecerahan lampu LED mengikut keperluan. Pengesan PIR disambungkan ke PORT B pada pengawalmikro PIC18F4550. Lampu LED disambungkan ke PORT C dan dikawal dengan PWM untuk meredupkan lampu. Bangunkan satu program dalam bahasa C menggunakan PIC18F4550 untuk mengawal kecerahan lampu berdasarkan gerakan yang dikesan oleh pengesan PIR dengan berpandukan Jadual B2. Dalam reka bentuk ini, sertakan bersama litar skematik bagi sistem kawalan lampu automatik garaj kereta.

Table B2: PWM Reference

| Condition/Stimulus | PIR Sensor (PORT B, RB0) | PWM Duty Cycle | LED Brightness |
|------------------------------------|--------------------------|----------------|-----------------|
| No motion detected (empty section) | LOW (0) | 0% | OFF |
| Motion detected (vehicles/people) | HIGH (1) | 12%=31 | Low brightness |
| Motion detected continuously | HIGH (1) | 48%=124 | Half brightness |

[20 marks]

[20 markah]

SOALAN TAMAT

TOCON: TIMER0 CONTROL REGISTER

| R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|--------|--------|-------|-------|-------|-------|-------|-------|
| TMR0ON | T08BIT | T0CS | T0SE | PSA | T0PS2 | T0PS1 | T0PS0 |
| bit 7 | | | | | | | bit 0 |

- bit 7 **TMR0ON:** Timer0 On/Off Control bit
 1 = Enables Timer0
 0 = Stops Timer0
- bit 6 **T08BIT:** Timer0 8-Bit/16-Bit Control bit
 1 = Timer0 is configured as an 8-bit timer/counter
 0 = Timer0 is configured as a 16-bit timer/counter
- bit 5 **T0CS:** Timer0 Clock Source Select bit
 1 = Transition on T0CKI pin
 0 = Internal instruction cycle clock (CLKO)
- bit 4 **T0SE:** Timer0 Source Edge Select bit
 1 = Increment on high-to-low transition on T0CKI pin
 0 = Increment on low-to-high transition on T0CKI pin
- bit 3 **PSA:** Timer0 Prescaler Assignment bit
 1 = Timer0 prescaler is not assigned. Timer0 clock input bypasses prescaler.
 0 = Timer0 prescaler is assigned. Timer0 clock input comes from prescaler output.
- bit 2-0 **T0PS2:T0PS0:** Timer0 Prescaler Select bits
 111 = 1:256 Prescale value
 110 = 1:128 Prescale value
 101 = 1:64 Prescale value
 100 = 1:32 Prescale value
 011 = 1:16 Prescale value
 010 = 1:8 Prescale value
 001 = 1:4 Prescale value
 000 = 1:2 Prescale value

T1CON: TIMER1 CONTROL

| U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|-----|---------|---------|---------|---------------------|--------|--------|
| — | — | T1CKPS1 | T1CKPS0 | T1OSCEN | $\overline{T1SYNC}$ | TMR1CS | TMR1ON |
| bit 7 | | | | | | | bit 0 |

- bit 7-6 **Unimplemented:** Read as '0'
- bit 5-4 **T1CKPS1:T1CKPS0:** Timer1 Input Clock Prescale Select bits
 11 = 1:8 prescale value
 10 = 1:4 prescale value
 01 = 1:2 prescale value
 00 = 1:1 prescale value
- bit 3 **T1OSCEN:** Timer1 Oscillator Enable Control bit
 1 = Oscillator is enabled
 0 = Oscillator is shut-off (the oscillator inverter is turned off to eliminate power drain)
- bit 2 **$\overline{T1SYNC}$:** Timer1 External Clock Input Synchronization Control bit
When TMR1CS = 1:
 1 = Do not synchronize external clock input
 0 = Synchronize external clock input
When TMR1CS = 0:
 This bit is ignored. Timer1 uses the internal clock when TMR1CS = 0.
- bit 1 **TMR1CS:** Timer1 Clock Source Select bit
 1 = External clock from pin RC0/T1OSO/T1CKI (on the rising edge)
 0 = Internal clock ($F_{osc}/4$)
- bit 0 **TMR1ON:** Timer1 On bit
 1 = Enables Timer1
 0 = Stops Timer1

T2CON: TIMER2 CONTROL

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

- bit 7 **RAPU:** PORTA Pull-up Enable bit
 1 = PORTA pull-ups are disabled
 0 = PORTA pull-ups are enabled by individual PORT latch values
- bit 6 **INTEDG:** Interrupt Edge Select bit
 1 = Interrupt on rising edge of RA2/INT pin
 0 = Interrupt on falling edge of RA2/INT pin
- bit 5 **T0CS:** Timer0 Clock Source Select bit
 1 = Transition on RA2/T0CKI pin
 0 = Internal instruction cycle clock (Fosc/4)
- bit 3 **PSA:** Prescaler Assignment bit
 1 = Prescaler is assigned to the WDT
 0 = Prescaler is assigned to the Timer0 module
- bit 2-0 **PS<2:0>:** Prescaler Rate Select bits

| BIT VALUE | TIMER0 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 |

Appendix 2

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 | PEIE | TOIE | INTE | RBIE ⁽¹⁾ | TOIF ⁽²⁾ | INTF | RBIF |
| 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: Global Interrupt Enable bit 1 = Enables all unmasked interrupts 0 = Disables all interrupts |
| bit 6 | PEIE: Peripheral Interrupt Enable bit 1 = Enables all unmasked peripheral interrupts 0 = Disables all peripheral interrupts |
| bit 5 | TOIE: Timer0 Overflow Interrupt Enable bit 1 = Enables the Timer0 interrupt 0 = Disables the Timer0 interrupt |
| bit 4 | INTE: INT External Interrupt Enable bit 1 = Enables the INT external interrupt 0 = Disables the INT external interrupt |
| bit 3 | RBIE: PORTB Change Interrupt Enable bit ⁽¹⁾ 1 = Enables the PORTB change interrupt 0 = Disables the PORTB change interrupt |
| bit 2 | TOIF: Timer0 Overflow Interrupt Flag bit ⁽²⁾ 1 = TMR0 register has overflowed (must be cleared in software) 0 = TMR0 register did not overflow |
| bit 1 | INTF: INT External Interrupt Flag bit 1 = The INT external interrupt occurred (must be cleared in software) 0 = The INT external interrupt did not occur |
| bit 0 | RBIF: PORTB Change Interrupt Flag bit 1 = When at least one of the PORTB general purpose I/O pins changed state (must be cleared in software) 0 = None of the PORTB general purpose I/O pins have changed state |

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 |

- 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 3

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/DONE | — | ADON |
| bit 7 | | | | | | bit 0 | |

bit 7-6 **ADCS1:ADCS0**: A/D Conversion Clock Select bits (ADCON0 bits in bold)

| ADCON1 <ADCS2> | ADCON0 <ADCS1:ADCS0> | Clock Conversion |
|-------------------|-------------------------|---|
| 0 | 00 | Fosc/2 |
| 0 | 01 | Fosc/8 |
| 0 | 10 | Fosc/32 |
| 0 | 11 | Frc (clock derived from the internal A/D RC oscillator) |
| 1 | 00 | Fosc/4 |
| 1 | 01 | Fosc/16 |
| 1 | 10 | Fosc/64 |
| 1 | 11 | Frc (clock derived from the internal A/D RC oscillator) |

bit 5-3 **CHS2:CHS0**: Analog Channel Select bits

000 = Channel 0 (AN0)
 001 = Channel 1 (AN1)
 010 = Channel 2 (AN2)
 011 = Channel 3 (AN3)
 100 = Channel 4 (AN4)
 101 = Channel 5 (AN5)
 110 = Channel 6 (AN6)
 111 = Channel 7 (AN7)

Note: The PIC16F873A/876A devices only implement A/D channels 0 through 4; the unimplemented selections are reserved. Do not select any unimplemented channels with these devices.

bit 2 **GO/DONE**: A/D Conversion Status bit

When ADON = 1;

1 = A/D conversion in progress (setting this bit starts the A/D conversion which is automatically cleared by hardware when the A/D conversion is complete)

0 = A/D conversion not in progress

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

bit 0 **ADON**: A/D On bit

1 = A/D converter module is powered up

0 = A/D converter module is shut-off and consumes no operating current

ADCON1 REGISTER (ADDRESS 9Fh)

| | | | | | | | |
|-------|-------|-----|-----|-------|-------|-------|-------|
| R/W-0 | R/W-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
| ADFM | ADCS2 | — | — | PCFG3 | PCFG2 | PCFG1 | PCFG0 |
| bit 7 | | | | bit 0 | | | |

bit 7 **ADFM:** A/D Result Format Select bit

1 = Right justified. Six (6) Most Significant bits of ADRESH are read as '0'.
0 = Left justified. Six (6) Least Significant bits of ADRESL are read as '0'.

bit 6 **ADCS2:** A/D Conversion Clock Select bit (ADCON1 bits in shaded area and in **bold**)

| ADCON1 <ADCS2> | ADCON0 <ADCS1:ADCS0> | Clock Conversion |
|-------------------|-------------------------|---|
| 0 | 00 | FOSC/2 |
| 0 | 01 | FOSC/8 |
| 0 | 10 | FOSC/32 |
| 0 | 11 | FRC (clock derived from the internal A/D RC oscillator) |
| 1 | 00 | FOSC/4 |
| 1 | 01 | FOSC/16 |
| 1 | 10 | FOSC/64 |
| 1 | 11 | FRC (clock derived from the internal A/D RC oscillator) |

bit 5-4 **Unimplemented:** Read as '0'

bit 3-0 **PCFG3:PCFG0:** A/D Port Configuration Control bits

| PCFG <3:0> | AN7 | AN6 | AN5 | AN4 | AN3 | AN2 | AN1 | AN0 | VREF+ | VREF- | C/R |
|---------------|-----|-----|-----|-----|-------|-------|-----|-----|-------|-------|-----|
| 0000 | A | A | A | A | A | A | A | A | VDD | VSS | 8/0 |
| 0001 | A | A | A | A | VREF+ | A | A | A | AN3 | 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 | AN3 | 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 | AN3 | VSS | 2/1 |
| 011x | D | D | D | D | D | D | D | D | — | — | 0/0 |
| 1000 | A | A | A | A | VREF+ | VREF- | A | A | AN3 | AN2 | 6/2 |
| 1001 | D | D | A | A | A | A | A | A | VDD | VSS | 6/0 |
| 1010 | D | D | A | A | VREF+ | A | A | A | AN3 | VSS | 5/1 |
| 1011 | D | D | A | A | VREF+ | VREF- | A | A | AN3 | AN2 | 4/2 |
| 1100 | D | D | D | A | VREF+ | VREF- | A | A | AN3 | AN2 | 3/2 |
| 1101 | D | D | D | D | VREF+ | VREF- | A | A | AN3 | AN2 | 2/2 |
| 1110 | D | D | D | D | D | D | D | A | VDD | VSS | 1/0 |
| 1111 | D | D | D | D | VREF+ | VREF- | D | A | AN3 | AN2 | 1/2 |

A = Analog input D = Digital I/O

C/R = # of analog input channels/# of A/D voltage references

ADCON2 REGISTER

| R/W-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|-----|-------|-------|-------|-------|-------|-------|
| ADFM | — | ACQT2 | ACQT1 | ACQT0 | ADCS2 | ADCS1 | ADCS0 |

bit 7

bit 0

- bit 7 **ADFM:** A/D Result Format Select bit
1 = Right justified
0 = Left justified
- bit 6 **Unimplemented:** Read as '0'
- bit 5-3 **ACQT2:ACQT0:** A/D Acquisition Time Select bits
111 = 20 TAD
110 = 16 TAD
101 = 12 TAD
100 = 8 TAD
011 = 6 TAD
010 = 4 TAD
001 = 2 TAD
000 = 0 TAD⁽¹⁾
- bit 2-0 **ADCS2:ADCS0:** A/D Conversion Clock Select bits
111 = FRC (clock derived from A/D RC oscillator)⁽¹⁾
110 = Fosc/64
101 = Fosc/16
100 = Fosc/4
011 = FRC (clock derived from A/D RC oscillator)⁽¹⁾
010 = Fosc/32
001 = Fosc/8
000 = Fosc/2