

Advanced Diploma Robotics and Mechatronics Engineering

(DMC - 52872WA)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Student full name:** | |  | | |
| I certify that the attached assessment is my own work and that any material drawn from other sources has been acknowledged.  Copyright in assessments remains my property, however  I grant permission to the Engineering Institute of Technology (EIT) to make copies of assessments for assessment, review and/or record keeping purposes.  I note that the EIT reserves the right to check my assessment for plagiarism.  Should the reproduction of all or part of an assessment be required by the EIT for any purpose other than those mentioned above, appropriate authorisation will be sought from me on the relevant form. | | | | |
| Please place a tick (☑) in the box below to indicate that you have read, understood, and certify the above statement.  Please include this page in/with your submission.  Any electronic responses to this submission will be sent to your Moodle account.  **AGREEMENT**       **DATE:** | | | | |
|  | | | | |
| Achieved / To be Achieved: | | | | |
| Assessor: |  | | Date: |  |
| Overall feedback: |  | | | |

**Guidelines for Students**

**How is this unit assessed?**

After completion of this assessment, you will be given a result of ‘Achieved’ or ‘To be Achieved’. The assessor will give you feedback via Moodle and you will have an opportunity to submit additional evidence if you have received a ‘To be Achieved’ result.

You will be allowed one (1) opportunity to resubmit the same assessment task, if required.

Once all assessment tasks for this unit have been completed, you will be given a final unit result of ‘Competent’ or ‘Not Yet Competent’. If you are deemed ‘Not Yet Competent’ in a unit after all resubmission attempts, you will be required to re-sit the unit.

**How is this assessment task assessed?**

For a result of ‘Achieved’ in this assessment, all unit elements (as indicated on page 4) must be completed to a satisfactory standard.

At Advanced Diploma level a ‘satisfactory’ standard, as stipulated by the Australian Qualifications Framework, means that you will demonstrate the application of knowledge and skills:

* with depth in areas of specialisation, in contexts subject to change
* with initiative and judgment in planning, design, technical or management functions with some direction
* to adapt a range of fundamental principles and complex techniques to known and unknown situations
* across a broad range of technical or management functions with accountability for personal outputs
* personal and team outcomes within broad parameters

Assessors also make decisions based on the following considerations:

* all parts of this assessment have been completed to a standard that satisfactorily meets the requirements set out in the assessment criteria (as per the unit outline).
* the assessment evidence provided is the student’s own work, except as appropriately acknowledged by the use of referencing.
* the evidence is recent and the student’s knowledge is up-to-date.

**Assessment Instructions:**

1. You must answer ALL questions.
2. Please ensure you complete your answers in a blue font (not red or black).
3. The best marks can be earned by giving concise, brief answers that address the questions.
4. You must reference all content used from other sources including course materials, slides, diagrams, etc. Do not directly copy and paste from course materials or any other resources.   
   Refer to the referencing section of the EIT eLibrary on Moodle for referencing guides.
5. Use this document for completing your answers by typing the answers after each question without deleting the question. Make sure that you preserve the original question number format.
6. Do not add extra pictures, etc. as annexures; instead, paste them directly into this answer sheet. Hand-drawn sketches can be inserted after scanning but please ensure that the file size does not become big (more than 10 MB). You must refer to all diagrams and pictures, etc. that you have drawn or pasted in.
7. When saving your document (must be Word format), ensure you include your name in the title: COURSECODE\_MODULE#\_ASSESSMENTTYPE\_VERSION#\_YOURNAME

**E.g. DMC\_DMCPIC613\_PracticalAssessment\_v1\_JohnSmith**

|  |  |
| --- | --- |
| **Unit no. and name:** | **DMCPIC613: Develop PIC programs** |
| **Assessment type:** | **Practical** |

Assessment Points:

* Supply the required answers below in blue font (not red or black).
* You must submit this assessment along with the written component.
* You must answer all questions.

PLEASE NOTE:

This practical requires that you use MPLAB and PICAXE software.

MPLAB

You can download MPLAB from:

<http://ww1.microchip.com/downloads/en/DeviceDoc/MPLAB_IDE_8_92.zip>

A newer version of MPLAB could be used:

<https://www.microchip.com/mplab/mplab-x-ide>

PICAXE

You can download PICAXE editor from:

<https://picaxe.com/software/picaxe/picaxe-editor-6/>

A multiplatform tool could alternatively be used:

<https://picaxe.com/software/picaxe/axepad/>

|  |  |  |  |
| --- | --- | --- | --- |
| **Q1** | You are developing a program in Assembly for a PIC16F84A that generates Pulse Width Modulation output to control a motor. The PWM cycle is 20ms. The duty cycle should be variable between 1ms and 19ms (5% to 95%).  The PWM signal should be produced on PortB pin 1.  The microcontroller is set to use an 800 kHz clock.  Timing precision of ±0.2ms is acceptable.   * Create the program in Assembly using MPLAB that sets the correct output at PortB and generates the PWM. The duty cycle could be any value that you insert in your code (it does not have to be obtained from an external output). The timing should be compatible with the clock frequency. * Provide a screenshot showing the code and the correct execution. * Provide a screenshot showing the stopwatch running when the output turns ON and then when it turns OFF (this should prove that the time between the pulses is 20ms)   *Hint: to manage the time you can create a loop that uses ‘nop’ and ‘goto’ for a certain amount of iterations. Recall that every instruction takes 4 clock cycles and that ‘goto’ takes 2 instructions while a ‘nop’ and other instructions take 1 instruction cycle.* | | |
| **A1** | **Student answer:** | | |
| **F1** | **Assessor feedback:** | Achieved |  |
| To be achieved |  |
|  |  | | |
| **Q2** | Write a program in Assembly for the PIC16F84A that generates an output sequence to control a stepper motor. The output sequence to generate is:  0001  ; wait for a certain amount of time  0010  ; wait for a certain amount of time  0100  ; wait for a certain amount of time  1000  ; wait for a certain amount of time  ; Repeat  The outputs are on pins 4 to 7 in PortB.  Write the program by ensuring that the waiting after each bit pattern is included.  Provide a screenshot showing the code and the correct execution with different values of PortB. | | |
| **A2** | **Student answer:** | | |
| **F2** | **Assessor feedback:** | Achieved |  |
| To be achieved |  |
|  | | | |  |
| **Q3** | Write the same program of Question 1 in PIC BASIC and provide a screenshot showing the execution when the output is ON and when the output is OFF.  Use PICAXE programming editor and select “PICAXE-18M2”. | | |
| **A3** | **Student answer:** | | |
| **F3** | **Assessor feedback:** | Achieved |  |
| To be achieved |  |
|  |  | | |
| **Q4** | You are designing a program for a robot controlled by a PIC microcontroller. The robot has a bump detector switch wired to one pin of the microcontroller. The robot should go forward until it hits an obstacle. As soon as the robot hits an obstacle (the bump switch sends an ON signal), the robot should stop, turn right by 90 degrees and then continue its forward motion.  You have to create two subroutines called “goforward” and “goright”. The “goforward” subroutine will run the robot forward for 500 ms and the “goright” subroutine turns the robot to the right by during 500 ms.  “goforward” has to set pins B.1 and B.2 to ON and pause for 500 ms before returning.  “goright” has set pin B.1 ON and in B.2 OFF and pause for 500 ms before returning.  Write the program using PICAXE programming editor and by selecting “PICAXE-18M2”. You have to configure an input that is connected to the bump switch. The input should be an interrupt-enabled one such as C.7.  Provide a screenshot showing the program code and the correct execution in the case where there is no interrupt and when there is an interrupt. | | |
| **A4** | **Student answer:** | | |
| **F4** | **Assessor feedback:** | Achieved |  |
| To be achieved |  |

**END OF ASSESSMENT**