



UNIVERSITY COLLEGE TATI (UCTATI)

FINAL EXAMINATION QUESTION BOOKLET

COURSE CODE	: BMT 4033
COURSE TITLE	: EMBEDDED SYSTEM DESIGN
SEMESTER/SESSION	: 2-20222023
DURATION	: 3 HOURS

Instructions:

1. This booklet contains 4 questions. Answer **all** questions.
2. All answers should be written in answer booklet.
3. Write legibly and draw sketches wherever required.
4. If in doubt, raise up your hands and ask the invigilator.

DO NOT OPEN THIS BOOKLET UNTIL YOU ARE TOLD TO DO SO

THIS BOOKLET CONTAINS 10 PRINTED PAGES INCLUDING COVER PAGE

QUESTION 1

- a) Describe the difference between python language with c language in developing embedded applications by giving **five (5)** comparisons. (5 marks)
- b) Describe for every line the meaning of every instruction.

```
from machine import Pin,PWM example: import function of Pin and PWM
from utime import sleep
```

```
led=Pin(3,Pin.OUT)
reed_sw= Pin(20, Pin.IN, Pin.PULL_UP)
buzzer=PWM((Pin(18)))
buzzer.deinit()
led.value(1)
while(1):
    if(reed_sw.value()==0):
        buzzer.deinit()
        led.value(1)
        sleep(0.3)
        led.value(0)
        sleep(0.8)
    else:
        buzzer.duty_u16(32767)
        buzzer.freq(500)
        led.value(1)
        sleep(0.1)
        led.value(0)
        sleep(0.2)
```

(20 marks)

QUESTION 2

a) Describe the difference between a microcontroller and a general-purpose microprocessor by giving **four (4)** characteristics that make these two devices different.

(4 marks)

b) Create an application that will turn on **eight LEDs** according to the following cases and repeat each case **four (4)** times:

- step 1: Turn on a single LED in one time from left to right.
- step 2: Turn on a single LED in one time from right to left.
- step 3: Blink two sets (4 left and 4 right sides) of LED alternately.
- step 4: blink all of 8 LEDs.

i. Produce schematic diagram for LED connection. Refer to figure 1 for the raspberry pi Pico pin out.

(8 marks)

ii. Outlines a program for this application.

(13 marks)

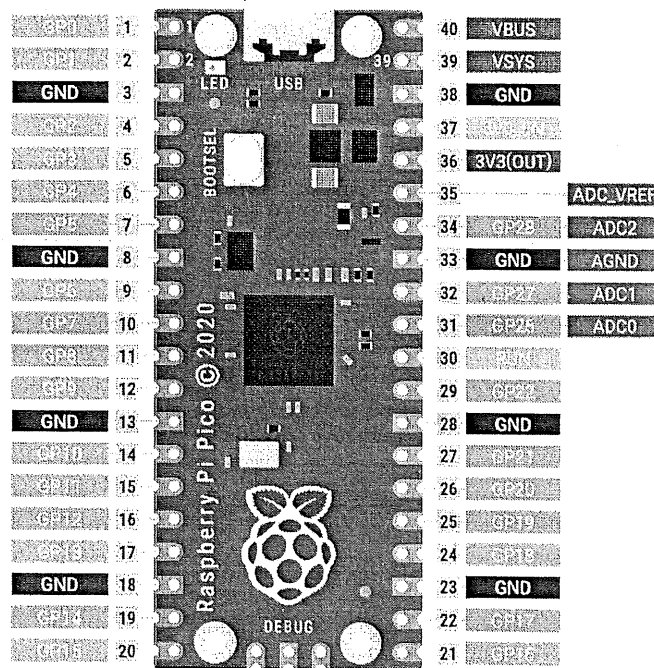


Figure 1: Pi Pico pin out

QUESTION 3

- a) Draw the internal construction of the 4x4 matrix keypad. (5 marks)
- b) Create an application that will give output at LCD 16x2 based on the keypad input value as follows:
- When the numbering value is pressed, the value is displayed on the second line at the center of the LCD.
 - When * is press, it will clear the LCD screen.
 - When either A, B, C, D or # is pressed, it will display a different message at the LCD screen at the first line of LCD.
Example: if A is pressed, it will display "welcome".
if B is pressed, it will display "thank you".
- i. Referring to the LCD and keypad in figure 3, produce the wiring diagram between these devices and the Pi Pico microcontroller. For Pi Pico pin out, refer to figure 1 in question 2. (8 marks)
- ii. Outline a program for this application. Refer attachment section for the LCD and keypad basic program as reference. (12 marks)

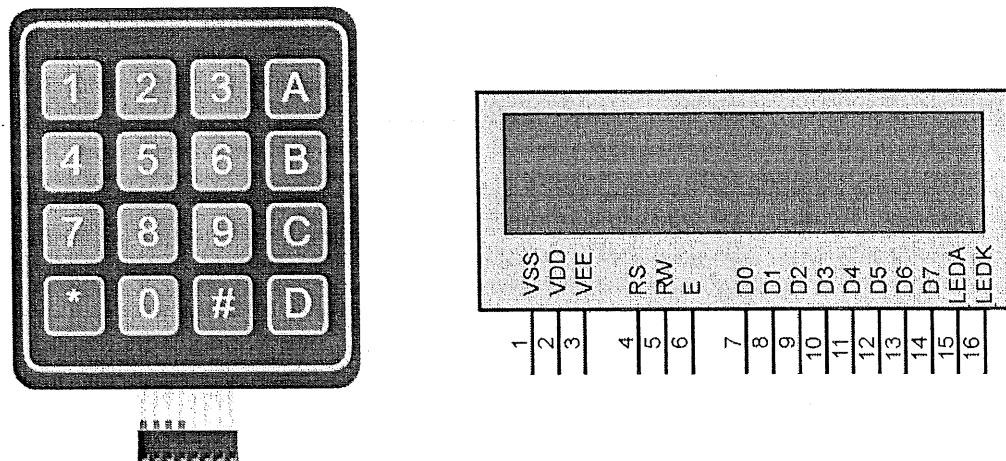


Figure 3: keypad and LCD

QUESTION 4

- a) State how to control the movement of DC servo motor by using PWM waveform. (5 marks)
- b) Figure 4 shows a structure of a fish robot who's the movement is controlled by two (2) servo motor at a fish tail:

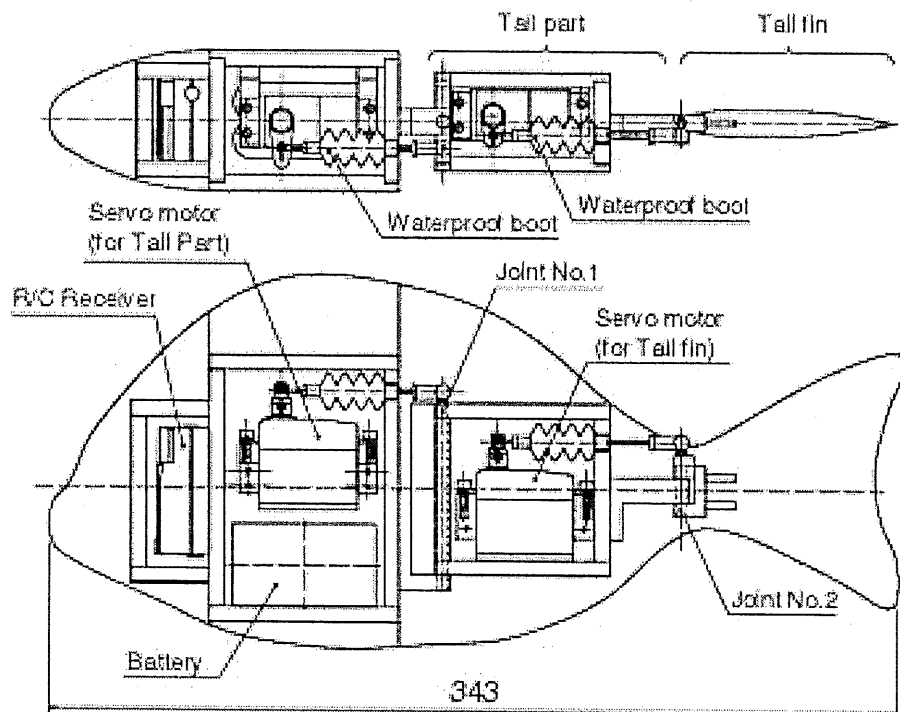


Figure 4: Fish robot Diagram

- i. Explain the servo motor operation to create a movement forward, turn left and turn right. (6 marks)
- ii. Outline a program that will make the fish robot to go forward 3 steps then left 3 steps and lastly right 4 steps. Then the movement is repeated. Refer attachment section for the basic servo motor program. (12 marks)

- iii. Produce a diagram how the servo motor is connected to Pi Pico in program. Use Figure 5 for servo reference for connection (2 marks)

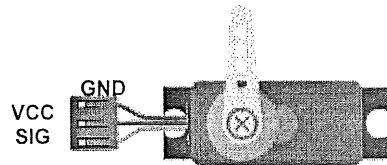


Figure 5: Servo Motor

-----End of question-----

BMT 4033 EMBEDDED SYSTEM DESIGN

Attachment:

LCD.py basic program routine example

```

from machine import Pin
from utime import sleep

rs=Pin(2,Pin.OUT)
e=Pin(3,Pin.OUT)
data=[Pin(4,Pin.OUT),Pin(5,Pin.OUT),Pin(6,Pin.OUT),Pin(7,Pin.OUT)]
      #   db4           db5           db6           db7

def write(nilai): #function to write value to lcd
    data[0].value((nilai & 0x10)>>4)    #send high nibble
    data[1].value((nilai & 0x20)>>5)
    data[2].value((nilai & 0x40)>>6)
    data[3].value((nilai & 0x80)>>7)
    e.value(0)        #enable low
    sleep(0.01)
    e.value(1)        #enable high
    sleep(0.01)

    data[0].value((nilai & 0x01)>>0)    #send low nibble
    data[1].value((nilai & 0x02)>>1)
    data[2].value((nilai & 0x04)>>2)
    data[3].value((nilai & 0x08)>>3)
    e.value(0)        #enable low
    sleep(0.01)
    e.value(1)        #enable high
    sleep(0.01)

def cmd_write(nilai): #function to write command
    rs.value(0)    #set rs 0
    write(nilai)
    sleep(0.01)

def data_write(nilai): #function to write data
    rs.value(1)    #set rs 1
    write(nilai)

def lcd_start():    #function to start lcd
    cmd_write(0x02)
    sleep(0.2)
    cmd_write(0x28)
    sleep(0.05)
    cmd_write(0x0c)
    cmd_write(0x01)
    cmd_write(0x80)

```

BMT 4033 EMBEDDED SYSTEM DESIGN

```

def print_lcd(string): #print at lcd
    for i in string:
        data_write(ord(i)) # pecah word ke satu2 character

def zfl(s, width):
    # Pads the provided string with leading 0's to suit the specified 'chrs'
    length
    # Force # characters, fill with leading 0's
    return '{:0>{w}}'.format(s, w=length)

```

Code (Hex)	Command to LCD Instruction Register
1	Clear display screen
2	Return home
4	Decrement cursor (shift cursor to left)
6	Increment cursor (shift cursor to right)
5	Shift display right
7	Shift display left
8	Display off, cursor off
A	Display off, cursor on
C	Display on, cursor off
E	Display on, cursor blinking
F	Display on, cursor blinking
10	Shift cursor position to left
14	Shift cursor position to right
18	Shift the entire display to the left
1C	Shift the entire display to the right
80	Force cursor to beginning to 1st line
C0	Force cursor to beginning to 2nd line
38	2 lines and 5x7 matrix

LCD command code

BMT 4033 EMBEDDED SYSTEM DESIGN

Keypad.py basic program routine example

```

from machine import Pin
from utime import sleep
#           row 1           row 2           row 3           row 4
key_row=[Pin(26, Pin.OUT),Pin(22, Pin.OUT),Pin(21, Pin.OUT),Pin(20,
Pin.OUT)]

#           col 1           col 2
key_col=[Pin(19, Pin.IN, Pin.PULL_UP),Pin(18, Pin.IN, Pin.PULL_UP),
#           col 3           col 4
Pin(17, Pin.IN, Pin.PULL_UP),Pin(16, Pin.IN, Pin.PULL_UP)]

key_label=['7','8','9','/',
           '4','5','6','x',
           '1','2','3','- ',
           '0','0','=','+']

def getkey(): # function to read keypad then return value pressed
    for i in range(4): # set all row to 1
        key_row[i].value(1)
    for i in range(4):
        key_row[i].value(0) # set row to 0
        for j in range(4):
            if(key_col[j].value()==0): #read
                sleep(0.1) # delay
                if(key_col[j].value()==0):#read again
                    while(key_col[j].value()==0): #wait key depressed
                        pass # pass while function
                    return key_label[j+i*4] #return key pressed
        key_row[i].value(1) # set row to 1
    return 0 # return no key pressed

```

BMT 4033 EMBEDDED SYSTEM DESIGN

Servo.py basic program routine example

```
from machine import Pin,PWM
min=500000 #0 degree angle
max=2400000 #180 degree angle
servo_1=PWM(Pin(27)) # servo connection
servo_1.freq(50) # 50hz to generate pulse 20ms

def servo_1_angle(angle): #angle from 0 to 180 degree
    duty=angle*(max-min)/180+min # convert time from angle value
    servo_1.duty_ns(int(duty)) # duty cycle time to servo
```