|  |
| --- |
| ICBRAMR  **School of Electronics Engineering (SENSE)**  **B. Tech – Electronics & Computer Engineering**  **BECE403E – EMBEDDED SYSTEM DESIGN**  **LAB RECORD**  **(L15+L16)**  **Submitted By**  **21BLC1228 – Mayukh Ray**    **Submitted To**  **Dr. Manoj Kumar Rajagopal**  **DATE: 7/02/2024**  **Slot:** L15+L16  **Date: 7/2/2024**  **LAB – 06: Working with Servo Motor & LDR**  **AIM:**  Implement and verify the logic on the STM32 Nucleo-64 board using Keil Studio Cloud IDE.  **Software Required:** ARM Keil Studio (Mbed Online Compiler)  **Hardware Required:** Micro USB cable, NUCLEO64-STM32L152 Board, LEDs, Jumper Wires (M-F and M-M), Breadboard  **Procedure:**   1. Go to ARM Keil Studio (<https://studio.keil.arm.com>) and log in 2. Select File → New → Mbed Project 3. Click the Example project drop-down list and select “mbed2-example-blinky” 4. In Project name field, provide the name of the new project and click Add project 5. Double click on the “main.cpp” file from the newly created project folder 6. Modify the code in the editor window as per the logic of your application 7. Check for any errors in the program under the “Problems” tab of the panels window 8. If no errors, connect the Nucleo Board to the computer using Micro USB Cable 9. Click Play icon (Run project) to upload and start the code execution on the board.   **PROGRAM:**  **Lab Task 1:** **Controlling the servo motor at one degree at a time**  Write a program to control the servo motor by rotating slowly from 0 degrees to 180 degrees, 45 degree at a time. When the motor has to be rotated 180 degrees, it will return to the initial position. Implement and verify this logic on STM32 board.  **Code:**  #include "mbed.h"  PwmOut PWM1(PC\_8);  int main() {  while(1){  PWM1.period\_ms(20);  PWM1.pulsewidth\_us(500);  wait(1);  PWM1.pulsewidth\_us(1000);  wait(1);  PWM1.pulsewidth\_us(1500);  wait(1);  PWM1.pulsewidth\_us(2000);  wait(1);  PWM1.pulsewidth\_us(2500);  wait(1);  }  }  **Output:**    **Output Verification:**    **Lab Task 2: Auto intensity street light controller**  Write a program to design auto intensity street light controller. This system helps the street light to get switched on automatically as per surrounding brightness. For example, sometimes when the weather become hazy its quite difficult to see anything then at that point this auto intensity street light gets switched on based on present lighting condition. Implement and verify this logic on STM32 board.  **Code:**  #include “mbed.h”  PwmOut PWM1(PC\_8);  AnalogIn Ain(PC\_3);  int main(){  while(1)  {  PWM1.period(0.010);  PWM1=1-Ain;  wait(0.1);  }  }  **Output:**      **Output Verification:**    **INFERENCE:**   1. We need to identify the pin connected to the LED. 2. Write the program to toggle the state of the LED between ON and OFF in a loop. 3. Compile and upload the code to the microcontroller.   **RESULT:**  **Lab-1:**  Hence, we were able to control servo motor by rotating from 0 degrees to 180 degrees, 45 degrees at a time. When the motor has to be rotated 180 degrees, it will return to the initial position.  **Lab-2:**  Hence, we were able to design auto intensity street light controller. This system helps the street light to get switched on automatically as per surrounding brightness. |