Internet of Things: Theory and Applications

Module 2: ESP32 based IoT Systems – Part B

Introduced by: Eng. Mohamed Hatem

Teaching Assistant at the Faculty of Computers and Data Science in Alexandria University

Graduate of Nanotechnology and Nano-electronics Engineering Program at the Zewail City of Sciences, Technology, and Innovation

Graduate Student Member at IEEE

Assignment – Module 2 – Part B - Recap

Internet of Things: Theory and Applications

ESP32 Project – Problem 1

- Design and implement a Home Access Control System using ESP32 microcontroller, a keypad, an IR sensor, SG90 servo motor, and an LDR sensor to provide secure access to a home. The system should allow users to input a password via the keypad, verify its correctness, the servo opens a door and then the system display real-time analog-based readings from the LDR and IR sensors on the serial monitor when the password is verified. The goal is to ensure that only authorized individuals can gain access to the home, and to provide the user with environmental data through the LDR and IR sensors for this authorized users only.
- **Specific Requirements:**
- **Keypad Input:**
  - The system must be able to accept a numerical password from the user via a 4x4 matrix keypad.
  - The password input must be masked for security purposes (e.g., asterisks or other symbols).

ESP32 Project – Problem 1

- **Password Verification:**
  - The entered password must be verified against a pre-defined correct password stored in the program code.
  - If the entered password matches the correct password, the system should display "Password Correct" on the serial monitor; otherwise, it should display "Password Incorrect."
- **Home Access Control – for Correct Passwords only:**
  - Upon successful password verification, the system should grant access to the home by activating an electronic lock means that the **servo motors goes from 0 to 180 then returns back.**
  - In the case of an incorrect password, access should be denied, and the user should be prompted to re-enter the password and **servo motor shall not move.**
- **LDR Sensor Integration:**
  - The system must be capable of reading real-time analog data from an LDR (Light Dependent Resistor) sensor.
  - The LDR sensor's analog readings (e.g., light intensity**) should be displayed on the serial monitor in a human-readable format if the password is correct only.**

ESP32 Project – Problem 1

- **IR Sensor Integration:**
  - The system must be able to read real-time analog data from an IR (Infrared) sensor.
  - The IR sensor's analog readings (e.g., proximity or distance) **should be displayed as analog value for the authorized users only.**
- **User-Friendly Interface:**
  - The system should have a user-friendly interface on the serial monitor, displaying clear instructions and status messages during the access control process.
  - ***The analog sensor readings should be displayed with appropriate units (e.g., Lux for LDR and centimeters for IR). - BONUS***

ESP32 Project – Problem 1

- **Bonus: Simulation & Schematic**
  - Design: Create a clear schematic diagram of the Smart Door system using Fritzing or similar software.
  - Simulation: Simulate the project on TinkerCAD to ensure proper functionality.

ESP32 Project – Problem 2

- Design an using ESP32 microcontroller, an IR sensor, LDR sensor, servo motor, push button, LCD, LED, and buzzer. The project aims to demonstrate real-world applications of sensors and actuators, offering a user-friendly interface via an LCD. When the push button is pressed, the system should display real-time readings of the IR and LDR sensors on the LCD. If the LDR sensor reading drops below a preset threshold, the system should trigger a visual and auditory alarm using the LED and buzzer. Additionally, when the push button is not pressed, the servo motor should execute a continuous back-and-forth motion between 0 and 180 degrees, with the angles being visible on the LCD.
- **Specific Requirements:**
- **ESP32 Integration:**
  - Utilize the ESP32 microcontroller to manage sensor readings, control the servo motor, and drive the LCD, LED, and buzzer.
  - Employ an appropriate development environment and libraries compatible with the ESP32 board for seamless implementation.

ESP32 Project – Problem 2

- Sensor Readings Display:
  - Upon pressing the push button, the system should display real-time analog readings from the IR sensor (e.g., distance or proximity) and the LDR sensor (e.g., light intensity) on the LCD.
  - Ensure the LCD presents sensor readings in an easily readable format with appropriate units.
- Low Light Alarm:
  - Continuously monitor LDR sensor readings when the push button is pressed.
  - When the LDR sensor reading falls below a predefined threshold value, activate an LED to illuminate and a buzzer to produce an audible alert, indicating a low light condition and an alarm shall be shown on the LCD.

ESP32 Project – Problem 2

- Servo Motion:
  - When the push button is not pressed, the system should initiate the servo motor to perform a smooth continuous back-and-forth motion between 0 and 180 degrees.
  - Display the current angle of the servo motor on the LCD during its movement.
- User Interaction:
  - The push button should act as the user interface trigger to switch between displaying sensor readings and activating the servo motion modes.
  - Pressing the push button should display the sensor readings, while releasing it should activate the servo motion.
- Safety and Stability:
  - Ensure the servo motor moves smoothly to avoid any abrupt movements or mechanical stress.

ESP32 Project – Problem 3 - BONUS

- Develop an advanced ESP32-based project that integrates an LCD, keypad, IR sensor, servo motor, buzzer, LED, push button, and LDR sensor to provide four distinct modes of operation. Upon pressing "1" on the keypad, the system should display real-time readings from the IR sensor and LDR sensor on the LCD, while triggering the LED and buzzer alarm when the LDR sensor reading falls below a threshold value. Pressing "2" on the keypad should initiate the servo motor to perform a scanning motion from 0 to 180 degrees and then back from 180 to 0 degrees. Mode "3" requires the user to input a password via the keypad, and if the password is correct, the system should display "Correct Password" on the LCD and enable the servo motor to move from 0 to 180 degrees; otherwise, it should display "Incorrect Password," and the servo should remain still. In mode "4," the system will operate as a smart door project, utilizing the analog IR sensor to detect objects or movement, and then activating the servo motor to operate the door accordingly and if a push button is pressed the IR sensor readings stops and the door holds on.

ESP32 Project – Problem 3 - BONUS

- **Specific Requirements:**
- ESP32 Integration:
  - Utilize the ESP32 microcontroller to control all components and manage the different modes of operation.
  - Ensure compatibility with appropriate development environments and libraries for smooth execution.
- LCD and Keypad Interface:
  - Implement a user-friendly interface on the LCD to display sensor readings, status messages, and password input prompts.
  - The keypad should enable users to select different modes by entering specific numbers (e.g., 1, 2, 3, 4).

ESP32 Project – Problem 3 - BONUS

- IR Sensor and LDR Integration:
  - Continuously read and display real-time analog data from the IR sensor and LDR sensor on the LCD when mode "1" is selected.
  - Trigger the LED and buzzer alarm when the LDR sensor reading falls below a predefined threshold in mode "1."
- Servo Motor Operation:
  - Enable the servo motor to perform a smooth scanning motion from 0 to 180 degrees and back from 180 to 0 degrees in mode "2."
  - In mode "3," the servo motor should move from 0 to 180 degrees if the entered password is correct; otherwise, it should remain stationary.
- Password Security:
  - Implement a secure password mechanism for mode "3," requiring users to input the correct password via the keypad to gain access.
  - Display "Correct Password" on the LCD and move the servo motor upon successful password entry; otherwise, display "Incorrect Password."

ESP32 Project – Problem 3 - BONUS

- Smart Door Functionality:
  - In mode "4," integrate the analog IR sensor to detect objects or movement in front of the smart door system.
  - Activate the servo motor to operate the smart door accordingly, allowing opening and closing in response to detected movement.
- Error Handling and Feedback:
  - Incorporate appropriate error handling mechanisms and provide informative feedback on the LCD to guide users during troubleshooting.

ESP32 Project – Problem 3 - BONUS

- Design a project that allows users to control the rotation of an SG90 servo motor to specific angles (e.g., 45°, 90°, 60°, 30°, 120°, 180°) using a keypad connected to an ESP32 microcontroller. The keypad should have buttons for each predefined angle, allowing users to select the desired angle for the servo motor's rotation.
- The system should include the following features:
- Keypad Interface: Interface a keypad with the ESP32 to receive user input for selecting the desired servo motor angle.
- Servo Motor Control: Connect the SG90 servo motor to the ESP32 and program it to rotate to the specified angle when the corresponding button on the keypad is pressed.
- Stop Button: Implement a "Stop" button on the keypad that halts the servo motor's movement at its current position.
- Reset Button: Include a "Reset" button that brings the servo motor back to the 0° position.
- BONUS: Custom Angle Button: Provide an additional button to allow users to input a custom angle (e.g., 18°) for the servo motor rotation from the serial monitor.

Bonus

- Design and implement a Smart Home Automation system using an ESP32 microcontroller that integrates various sensors, including an LDR (Light Dependent Resistor), piezo buzzer, an analog IR sensor, and a potentiometer, along with a SG90 servo motor and an LED. The system aims to automate window control based on ambient light conditions and detect the presence of a person to provide enhanced security and convenience.
- **Requirements:**
- **LDR Integration:** Interface the LDR with the ESP32 to detect ambient light conditions. When it detects darkness (low light intensity), the system should activate the servo motor to open the window to its maximum (180 degrees). Conversely, when the LDR detects light (high light intensity), the servo motor should close the window and return to the 0-degree position.
- **IR Sensor Integration:** Connect the IR sensor to the ESP32 to detect the presence of a person. When a person is detected, the system should light up the LED and run the buzzer to indicate the presence and simultaneously open the servo motor to the 180-degree position. When no person is detected, the servo motor should return to the 0-degree position and switch off the LED and stops the buzzer.
- **Potentiometer Control:** Utilize a potentiometer to provide manual control over the servo motor's angle. The user can rotate the potentiometer to adjust the window position within the range of 0 to 180 degrees.
- **Double Bonus:** Add a reset push button that get the servo to 0 degrees, switch off the LED, and the buzzer when pressed.

Double Bonus

- Design and implement a Smart Home Automation system using the ESP32 module, integrating various sensors, including an MQ4 gas sensor, a PIR motion sensor, a digital temperature and humidity sensor, along with an SG90 servo motor, an LED, a buzzer, and an LCD display. The system aims to monitor methane gas concentration, display temperature and humidity readings, detect motion, and provide enhanced security and alerts in case of any intrusion detected by the motion sensor.
- **Requirements:**
- **MQ4 Gas Sensor:** Interface the MQ4 gas sensor with the ESP32 to detect methane gas concentration. Display the real-time gas concentration value on the LCD and run LED and Buzzer alarm on any assumed threshold value.
- **Digital Temperature and Humidity Sensor:** Connect the digital temperature and humidity sensor to the ESP32 to measure and display temperature and humidity readings on the LCD and run LED and Buzzer alarm on any assumed threshold value.
- **Motion Sensor:** Integrate the motion sensor with the ESP32 to detect any human presence. When motion is detected, display "Motion Detected" on the LCD for a short time and move the SG90 servo motor to a predefined position as a smart door then run a LED and buzzer alarm for this intrusion for short time and stop them.
- **Servo Motor Control:** Program the ESP32 to control the SG90 servo motor's movement in response to motion sensor inputs and return it to its initial position after a specified time.
- **LED and Buzzer Alarm:** Implement code on the ESP32 to activate the LED and buzzer as an alarm when motion is detected, alerting occupants of possible intrusion, upon threshold methane gas concentration detected, and threshold temperature and humidity sensor detected.

Double Bonus

- Design and implement a Smart Home Automation system using the ESP32 module, integrating various sensors, including a Flame sensor, an Ultrasonic sensor, a digital temperature and humidity sensor, along with an SG90 servo motor, an LED, a buzzer, and an LCD display. The system aims to monitor flame detection, measure temperature and humidity, detect obstacles using the Ultrasonic sensor, and provide enhanced security and alerts in case of critical temperature and humidity values.
- **Requirements:**
- **Flame Sensor:** Interface the Flame sensor with the ESP32 to detect flames or high-temperature sources. When a flame is detected, activate the LED and buzzer as an alarm for immediate response and send a warning message to the LCD.
- **Digital Temperature and Humidity Sensor:** Connect the digital temperature and humidity sensor to the ESP32 to measure and display real-time temperature and humidity readings on the LCD and run a threshold LED and buzzer alarm on any assumed threshold values.
- **Ultrasonic Sensor:** Integrate the Ultrasonic sensor with the ESP32 to detect obstacles or proximity to objects. Display the distance measurements on the LCD, and open and close servo based on a threshold value detected along with running LED and buzzer alarm when this threshold value exceeded and print a warning message on the LCD and stop real time data when this happen.
- **Servo Motor Control:** Program the ESP32 to control the SG90 servo motor's movement based on the Ultrasonic sensor's distance measurements.
- **LED and Buzzer Alarm:** Implement code on the ESP32 to activate the LED and buzzer as an alarm when a flame is detected or when critical temperature and humidity values are measured or threshold ultrasonic sensor values is detected.

Triple Bonus - Design of Autonomous Wheel Robot with Sensing and Avoidance Capabilities

- Using the following component design and implement a autonomous wheel robot with sensing and avoidance capabilities
- **ESP32:** The brain of the robot, controlling various functions, sensors, and actuators.
- **Chassis:** A robot platform with two wheels and a cluster wheel for stability.
- **Motors:** Two DC motors for wheel propulsion.
- **L298N Motor Driver:** To control and drive the DC motors.
- **Power Source:** Power bank for the main power supply and a separate 9V battery for the L298N motor driver and control panel (CP) to manage power efficiently.
- **Flame Sensor:** To detect flames or high-temperature sources.
- **Methane Gas Sensor:** To detect methane gas concentration.
- **Digital Temperature and Humidity Sensor:** To measure temperature and humidity.
- **Ultrasonic Sensor:** For obstacle avoidance and distance measurements.
- **Servo Motor:** To rotate and scan for obstacles.
- **LCD Display:** To show real-time sensor readings and system status.
- **LED and Buzzer:** For alarm notifications.
- **Keypad:** For user input and robot control.

Triple Bonus - Design of Autonomous Wheel Robot with Sensing and Avoidance Capabilities

- **Functionalities**:
- Movement: The robot moves on its two wheels and cluster wheel using the two DC motors, controlled by the L298N motor driver.
- Sensing: The robot constantly reads values of methane concentration, flame, temperature, and humidity using their respective sensors every second.
- Display: The LCD display shows the real-time sensor readings, including methane concentration, flame detection, temperature, and humidity.
- Alarm: If any of the sensor readings exceed predetermined threshold values, an LED and buzzer alarm will be triggered to notify the user.
- Obstacle Avoidance: The robot uses an Ultrasonic sensor and a SG90 Servo motor to detect obstacles and avoid collisions by rotating and scanning its surroundings.
- User Control: The robot can be controlled using a keypad attached to the control panel (CP). Upon user input, the robot can stop, rotate to four different angles at various speeds, and perform other actions.

Triple Bonus - Design of Autonomous Wheel Robot with Sensing and Avoidance Capabilities

- **Power Management:**
- The main power source is a power bank, providing power to the main robot components.
- To manage power more efficiently, the L298N motor driver and CP are powered separately using a 9V battery, allowing independent control of motor and sensing systems.
- **Restart Capability:**
- The robot has a restart button located on its back. When pressed, it will restart all systems, providing a quick and convenient way to reset the robot if needed.
- **Safety Considerations:**
- The robot should be designed with safety features, such as proper insulation and casing for electrical components, to avoid any hazards.
- All sensor readings should be carefully monitored to prevent hazardous situations, especially when dealing with methane gas and flames.