LAB REPORT

MCTA 3203

SECTION 1

GROUP D

EXPERIMENT 4B

Serial and USB interfacing with microcontroller and computer-based system (2): Sensors and actuators

MATIC NO.	NAME
2111633	MUHAMMAD FARID BIN JAFRI
2110765	MUHAMMAD ARIF SYAZWAN BIN MOHD RIZAL
2112969	MUHAMMAD HAZIQ HAIKAL BIN SUAIB
2115207	MUHAMMAD AMIRUL SYAFIQ BIN RUSLANI
2116281	MUHAMMAD FADHLUL WAFI BIN AHMAD NAIM

SUBMISSION DATE: 8/11/2023

INSTRUCTOR: DR. WAHJU SEDIONO

Abstract

The main target of this lab report is to give the opportunity to students on how RFID in toll works. This lab report describes how to connect an RFID card reader to a computer. After connecting the RFID card reader to the computer, the students can control the servo motor and RFID card authentication by using Python and Arduino. In addition, RFID card readers that are USB-connected usually function as USB Human Interface Devices (HID), which means that in order to enable USB HID communication, a specific library or module must be used. The report details the materials required to set up this experimental framework, including RFID card readers, Arduino boards, servo motors, and other components essential for a successful lab experiment.

In summary,this experiment consists of an outline of the lab's objectives, methodology, and main elements. This lab experiment also gives the opportunity to the students on how to control and conduct the toll systems by using RFID and servo.This will enhance the knowledge gained by the students from time to time. Lastly, it provides context so that readers can grasp the primary goals and techniques used.

Table of Contents

Introduction

This week,we need to do Serial and USB interfacing with a microcontroller and computer-based system (2): Sensors and actuators. The goal of this project is to connect the RFID card reader to the computer by using USB and code it with Arduino and Python so that the output can be shown in the serial monitor whether the card has access or is denied. This experiment also required a unique approach to communication, by involving a specific library or module.

Our lab experiment explores the practical setup for RFID card authentication and servo motor control using Python and an Arduino microcontroller. In order to achieve this, we need a variety of supplies, such as servo motors, Arduino boards, RFID card readers, and other essential parts that work together to build an adaptable system for a variety of uses. This lab report offers strategies for bridging the hardware-software gap and producing better results and solutions on how RFID card readers related to servo motors and LED. This lab report basically demonstrates how the RFID works in real life like in the tolls.

Materials and Equipment

Materials Needed:

1. Arduino Board
2. RFID card reader with USB connectivity
3. Jumper Wires
4. RFID tags/cards
5. Servo motors
6. Breadboard
7. USB cables
8. Power supply
9. Different coloured LEDs

Equipment:

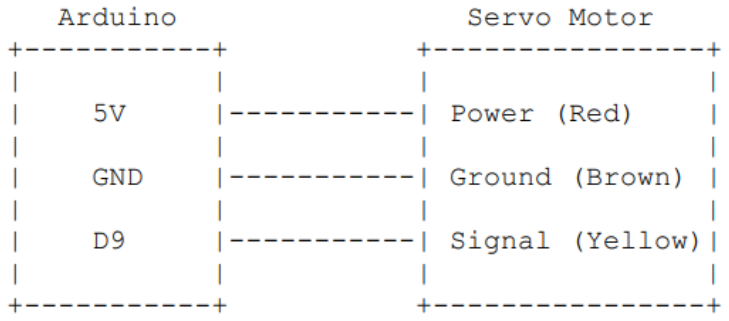
1. Computer with necessary software for programming and testing (Arduino IDE and python)

Experimental Setup

Hardware Setup:

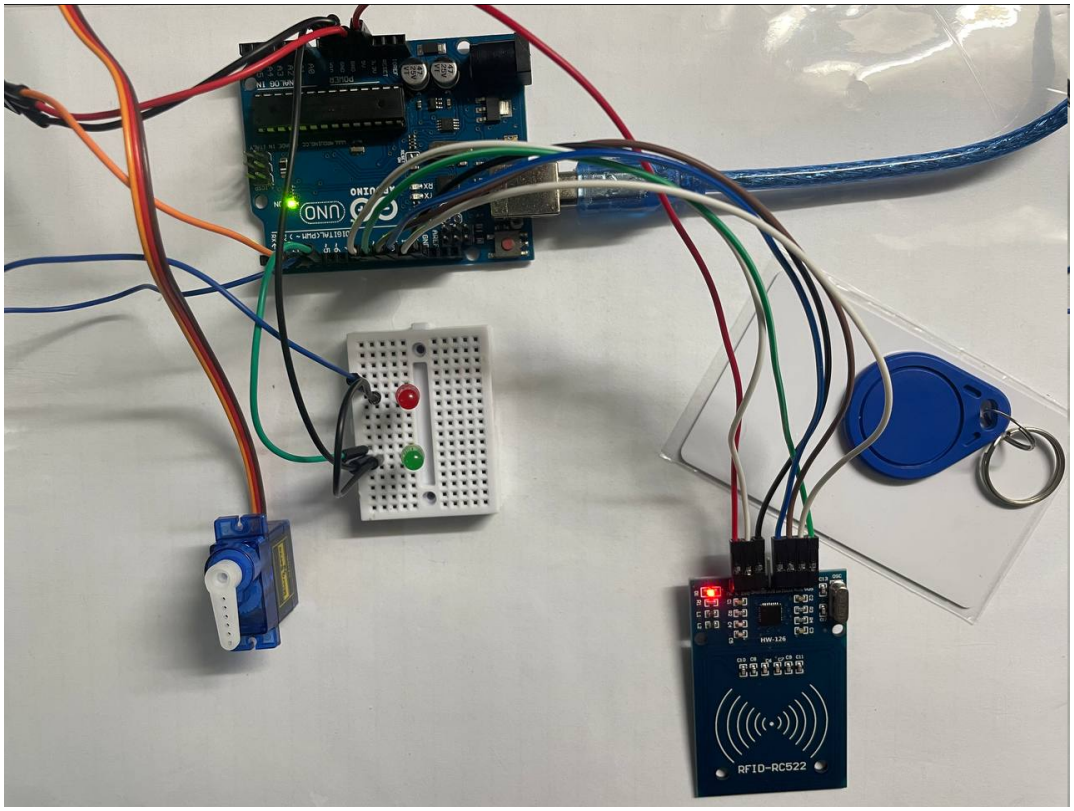
Wiring a Servo Motor:

1. The servo's power wire (red) was connected to the 5V output on the Arduino.
2. The servo's ground wire (brown) connected to one of the ground (GND) pins on the Arduino.
3. The servo's signal wire (yellow) was connected to one of the PWM pins on the Arduino (e.g., pin 9).
4. A common ground between the Arduino and the servo motor was connected to complete the circuit.



RFID Card Reader:

1. Power (VCC): The RFID reader's power pin should have been connected to the appropriate voltage source (e.g., 3.3V) on the Arduino. It was important to provide the correct voltage based on the reader's specifications.
2. Ground (GND): The RFID reader's ground pin was connected to the ground (GND) on the Arduino to establish a common ground reference.
3. The RFID reader's data output pin (typically labelled as TX or SDA) was connected to a digital pin on the Arduino, which was used for data communication.
4. Data (RX or SCL): If your RFID reader required two-way communication (e.g., for configuration or advanced features), you might have needed to connect the reader's data input pin (often labelled as RX or SCL) to another digital pin on the Arduino.


UNO
RFID-RC522

Methodology

Procedures:

1. A complete circuit including RFID and servo motor were set up as in the figure above.
2. The Arduino code has been verified and uploaded into the Arduino microcontroller using the Arduino IDE.
3. The tag number of the RFID card has been found in the serial monitor of the Arduino IDE.
4. The python code was saved into the computer file and run to observe the output.

Coding:

```
#include <SPI.h>
#include <MFRC522.h>
#include <Servo.h>

#define SS_PIN 10
#define RST_PIN 9
#define LED_G 5
#define LED_R 4
MFRC522 mfrc522(SS_PIN, RST_PIN);
Servo myServo;

void setup()
{
  Serial.begin(9600);
  SPI.begin();
  mfrc522.PCD_Init();
  myServo.attach(3);
  myServo.write(0);
  pinMode(LED_G, OUTPUT);
  pinMode(LED_R, OUTPUT);
}
void loop()
{
  if ( ! mfrc522.PICC_IsNewCardPresent())
  {
    return;
  }
  if ( ! mfrc522.PICC_ReadCardSerial())
  {
    return;
  }
  //Show UID on serial monitor
  //Serial.print("UID tag :");
  String content= "";
  byte letter;
  for (byte i = 0; i < mfrc522.uid.size; i++)
  {
     content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
     content.concat(String(mfrc522.uid.uidByte[i], HEX));
  }
```

```
  content.toUpperCase();
  if (content.substring(1) == "43 A6 9F 95")
  {
    Serial.println("A");
    delay(500);
    digitalWrite(LED_G, HIGH);
    myServo.write(90);
    delay(5000);
    myServo.write(0);
    digitalWrite(LED_G, LOW);
  }
  else   {
    Serial.println("D");
    digitalWrite(LED_R, HIGH);
    delay(1000);
    digitalWrite(LED_R, LOW);
  }
}
```

1. Arduino

```
import serial
import time

ser = serial.Serial('COM5', 9600)
print("Put your card to the reader...")

def read_rfid():
    ser.write(b'C')
    time.sleep(0.1)
    return ser.readline().decode().strip()

try:
    while True:
        rfid_data = read_rfid()
        if rfid_data == 'A':
            print("Access granted.")
        elif rfid_data == 'D':
            print("Access denied.")
        else:
            rfid_data = read_rfid()

except KeyboardInterrupt:
    print("Exiting the program.")

ser.close()
```

2. Python

* These codes can be found on our GitHub.

Results (Observation)

The authorised RFID is defined in our code. When the recognized UID is detected by the RFID reader, the servo motor will be adjusted to 180 degrees to indicate that the gate is open and green LED illuminated for visual indicator as per ask in the task given. After a few seconds of delay, the servo motor is adjusted to its original position, 90 degrees to indicate that the gate is closed again. When an unrecognised card/ UID is read, the servo motor does not move and remains at its original position, 90 degrees. And the red LED illuminated for visual indicator, indicates that the card/ UID is unrecognisable and the gate would not open.

Link for the video:
https://drive.google.com/file/d/1PzraQgltd6TnmAyHYTx4gzOZ-WU4lSfZ/view?usp=sharing

Discussion

The attempt to utilise the USB RFID scanner provided for reading card RFIDs was unsuccessful. We encountered problems where the computer couldn't detect the RFID scanner, even after trying different scanners and buying a new cable. This issue was encountered in other groups as well. We attempted using various codes from different websites, but unfortunately, there was still no data output from the RFID scanner to the computer even though it can scan the card and all. As a result, we were unable to employ the RFID cards and scanners for this experiment using the given equipment. Instead, we borrowed the different model of RFID from another group, soldered and studied its connection and finally it works as we intended it to be.

Conclusion

In conclusion, the enhanced code successfully integrates visual indicators using green and red LEDs to signal the recognition status of UID cards detected by the RFID reader. The introduction of structured JSON data handling not only improves the organisation of the code but also enhances its flexibility, allowing for easier modification and expansion of the system. The addition of user-configurable options for setting the angle position of the servo adds a valuable layer of customization, catering to varying user preferences and specific application requirements. This updated code not only enhances the functionality of the RFID-based system but also provides a more user-friendly and adaptable solution for UID recognition and servo control.

Recommendations

To further enhance the capabilities and usability of the system, several recommendations can be implemented. Firstly, it is advisable to strengthen the code's resilience by incorporating a robust error-handling mechanism to gracefully manage unexpected scenarios. Adding security measures, such as encryption, would bolster the protection of UID data and prevent unauthorised access. Improving the user interface with clear feedback mechanisms or a display would enhance the user experience, providing real-time information on system status and recognition outcomes.

Next, integrating logging functionality and analytics tools could offer valuable insights into system performance and usage patterns. Enabling remote configuration options would provide users with greater flexibility in adjusting settings without requiring direct physical access to the device.

References

1. Hercog, D., & Gergič, B. (2014, June 2). A Flexible Microcontroller-Based Data Acquisition Device. Sensors; Multidisciplinary Digital Publishing Institute. https://doi.org/10.3390/s140609755
2. *Sensor / Actuator interfacing through serial port*. (2009, October 12). Arduino Forum. https://forum.arduino.cc/t/sensor-actuator-interfacing-through-serial-port/4039
3. Abba, I. V. (2023, October 5). *The Arduino Handbook – Learn Microcontrollers for Embedded Systems*. freeCodeCamp.org. https://www.freecodecamp.org/news/the-arduino-handbook/

Student's Declaration

Certificate of Originality and Authenticity

This is to certify that we are responsible for the work submitted in this report, that the original work is our own except as specified in the references and acknowledgment, and that the original work contained herein has not been untaken or done by unspecified sources or persons.

We hereby certify that this report has not been done by only one individual and all of us have contributed to the report. The length of contribution to the reports by Each individual is noted within this certificate.

We also hereby certify that we have read and understand the content of the total report and no further improvement on the reports is needed from any of the individual contributor to the report.

We, therefore, agreed unanimously that this report shall be submitted for marking and this final printed report has been verified by us

Signature: *arifsyazwan* — Read✔️
Name: Muhammad Arif Syazwan Bin Mohd Rizal — Understand ✔️
Matric Number:2110765 — Agree ✔️

Signature: *haziqhaikal* — Read ✔️
Name: Muhammad Haziq Haikal bin Suaib — Understand ✔️
Matric Number: 2112969 — Agree ✔️

Signature: *amirul* — Read ✔️
Name: Muhammad Amirul Syafiq Bin Ruslani — Understand✔️
Matric Number: 2115207 — Agree✔️

Signature: *Wafi* — Read ✔️
Name: Muhammad Fadhlul Wafi Bin Ahmad Naim — Understand ✔️
Matric Number: 2116281 — Agree ✔️

Signature: *faridjafri* — Read ✔️
Name: Muhammad Farid Bin Jafri — Understand ✔️
Matric Number: 2111633 — Agree ✔️