PRACTICAL REPORT

FOR IOT PRACTICAL

2022

DARSHAN RAMJIYANI (DSP)

DOCS, KSKV Kachchh University

❖ 5.E1 – Simple Analog and Digital Input

Create an ATM password verification clone using arduino and **4x4 Keypad**.

Arduino Code:

```
const int baudRate = 9600;
const int passwordLength = 10;
const int keypadRows = 4;
const int keypadColumns = 4;
const char keypadKeys[keypadRows][keypadColumns] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
const int rowPins[keypadRows] = {12, 11, 10, 9};
const int colPins[keypadColumns] = {8, 7, 6, 5};
char passwordChars[passwordLength];

void setup() {
  Serial.begin(9600);
  for (int row = 0; row < keypadRows; row++)
  {
    pinMode(rowPins[row],INPUT);
    digitalWrite(rowPins[row],HIGH);
  }
  for (int column = 0; column < keypadColumns; column++)
  {
    pinMode(colPins[column],OUTPUT);
    digitalWrite(colPins[column],HIGH);
  }
  pinMode(LED_BUILTIN, OUTPUT);
  digitalWrite(LED_BUILTIN, HIGH);
  setPassword();
}


void loop()
{
  Serial.print("[Arduino] : Enter password to continue :- ");
  int charEntered = 0;
  char passwordEntered[passwordLength];

  while(charEntered < passwordLength)
  {
    char keyPressed = getPressedKey();
```

```
    if(keyPressed != '\0')
    {
      passwordEntered[charEntered] = keyPressed;
      Serial.print("*");
      charEntered += 1;
    }
  }
  Serial.println();
  if(passwordLength == charEntered)
  {
    if(matchPassword(passwordEntered))
    {
      Serial.println("[Arduino] : Password Matched. Welcome to the ATM.");
    }
    else
    {
      Serial.println("[Arduino] : Password doesn't match.");
    }
  }
  else
  {
    Serial.println("[Arduino] : Password matching failed. Try to restart the program.");
  }
  delay(10000);
}

void setPassword()
{
  Serial.println("[Arduino] : Please create password first.");
  Serial.print("[Arduino] : Enter your ");
  Serial.print(passwordLength);
  Serial.print(" digit password :- ");

  int numCharPasswordEntered = 0;

  while(numCharPasswordEntered < passwordLength)
  {
    char keyPressed = getPressedKey();

    if(keyPressed != '\0')
    {
      passwordChars[numCharPasswordEntered] = keyPressed;
      Serial.print("*");
      numCharPasswordEntered += 1;
    }
  }
  Serial.println();
```

```
  if(passwordLength == numCharPasswordEntered)
  {
    Serial.println("[Arduino] : Password has been created successfully.");
  }
  else
  {
    Serial.println("[Arduino] : Password creation has been failed! Restart
program to solve the issue.");
  }
}

char getPressedKey()
{
  char key = '\0';

  for(int i = 0; i < keypadColumns; i++)
  {
    digitalWrite(colPins[i],LOW);
    for(int j = 0; j < keypadRows; j++)
    {
      if(digitalRead(rowPins[j]) == LOW)
      {
        delay(20);
        while(digitalRead(rowPins[j]) == LOW);
        key = keypadKeys[i][j];
      }
    }
    digitalWrite(colPins[i],HIGH);
  }
  return key;
}

bool matchPassword(char enteredPassword[passwordLength])
{
  bool isMatch = true;

  for(int i=0; i < passwordLength; i++)
  {
    if(enteredPassword[i] != passwordChars[i])
    {
      isMatch = false;
    }
  }

  return isMatch;
}
```

Output / Circuit Diagram: