UNIVERSIDAD TECNOLÓGICA DE TAMAULIPAS NORTE

Facultad de Tecnologías de la Información

*T.S.U en Tecnologías de la Información Área Desarrollo De Software Multiplataforma*

**Formato para prácticas**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Subject** | | | **Quarter-month period** | | **Group** | **Period** | |
| Aplicaciones IoT | | | 5to | | A | May-Ago | |
|  | |  | |  | |  | |
| **No.** | **Practice** | | | | | | **Duration** |
| 1 | Using the DHT11 Sensor to Measure Temperature with Arduino | | | | | | 1 hr |
|  | |  | |  | |  | |
| 1. **Introduction**   In this practice, we will be exploring the use of the DHT11 sensor to measure temperature using the Arduino microcontroller. The DHT11 sensor is a low-cost, digital temperature and humidity sensor that is commonly used in a variety of projects, such as weather stations, thermostats, and environmental monitoring systems. By the end of this practice, you will have a better understanding of how the DHT11 sensor works and how to integrate it with Arduino to measure temperature. | | | | | | | |
|  | | | | | | | |
| 1. **Description** | | | | | | | |
| 1. **Materials needed**  * Arduino Uno board * DHT11 temperature and humidity sensor * Breadboard * Jumper wires * USB cable * Computer with Arduino IDE installed | | | | **Other** | | | |
| 1. **Practice development**   The DHT11 sensor is just one example of the many sensors that can be used with Arduino to measure various physical phenomena, such as light, sound, pressure, and motion. By learning how to use sensors with Arduino, you can develop a wide range of skills and knowledge that will be valuable in many fields, such as engineering, physics, and environmental science.  The DHT11 sensor is one of the most used sensors in Arduino projects due to its low cost and ease of use. This sensor can measure temperature with an accuracy of ±2°C and humidity with an accuracy of ±5%, and it is ideal for monitoring temperature and humidity in indoor environments, such as homes, offices, and classrooms.  Now, let’s start the practice:   1. Connect the DHT11 sensor to the breadboard and the Arduino Uno board as follows: 2. Connect the VCC pin of the DHT11 sensor to the 5V pin on the Arduino Uno board. 3. Connect the GND pin of the DHT11 sensor to the GND pin on the Arduino Uno board. 4. Connect the data pin of the DHT11 sensor to digital pin 2 on the Arduino Uno board. 5. Open the Arduino IDE on your computer and create a new sketch. 6. Copy and paste the following code into your sketch:   #include <DHT.h>  #define DHTPIN 2  #define DHTTYPE DHT11  DHT dht(DHTPIN, DHTTYPE);  void setup() {  Serial.begin(9600);  dht.begin();  }  void loop() {  delay(2000);  float temperature = dht.readTemperature();  Serial.print("Temperature: ");  Serial.print(temperature);  Serial.println(" \*C");  }   1. Verify and upload the sketch to the Arduino Uno board. 2. Open the Serial Monitor in the Arduino IDE to view the temperature readings from the DHT11 sensor. 3. Experiment with different environmental conditions to see how the temperature readings change.   -------------------- LM35 SENSOR -------------------------   1. Connect the LM35 to the breadboard, making sure the pins are aligned with the rows on the breadboard. 2. Connect the LM35's VCC pin to the Arduino's 5V pin, the GND pin to the Arduino's GND pin, and the output pin to an analog input pin on the Arduino (e.g. A0). 3. Open the Arduino IDE on your computer and create a new sketch. 4. Define a variable to store the LM35 output value using the analogRead() function. 5. Convert the analog value to a temperature value using the following formula: temperature = (analogValue \* 5) / 1023 \* 100. The LM35 outputs a voltage proportional to the temperature, with a sensitivity of 10mV per degree Celsius. 6. Print the temperature value to the Serial Monitor using the Serial.println() function. 7. Upload the sketch to the Arduino board. 8. Open the Serial Monitor to see the temperature readings.   // Declare the LM35 sensor pin  const int lm35Pin = A0;  void setup() {  // Initialize serial communication  Serial.begin(9600);  }  void loop() {  // Read the analog value from the LM35 sensor  int lm35Value = analogRead(lm35Pin);  // Convert the analog value to temperature in Celsius  float temperature = (lm35Value \* 5.0) / 1023.0 \* 100.0;  // Print the temperature to the serial monitor  Serial.print("Temperature: ");    // We take 50 from temperature to measure accurately (Tinkercad problem)  Serial.print(temperature-50);  Serial.println(" C");  // Delay for 1 second before taking another reading  delay(1000);  }  -------------------------------------------------------------------  **Follow-up questions:**   1. What is the range of temperature and humidity that the DHT11 can measure accurately?   **DHT11 is designed to operate accurately in a temperature range of 0°C to 50°C (32°F to 122°F) and a relative humidity range of 20% to 90%.**   1. Can you describe the process of how the DHT11 sensor measures temperature and humidity? 2. **When the sensor is powered on, it first takes a brief moment to stabilize. It then sends a start signal to the device it is connected to. The sensor then waits for a response from the device before transmitting data.** 3. **To measure temperature, the thermistor detects changes in resistance as the temperature changes. The sensor converts the resistance into a temperature reading, which is transmitted to the device.** 4. **To measure humidity, the capacitive humidity sensor detects changes in capacitance as humidity levels change. The sensor converts the capacitance into a humidity reading, which is also transmitted to the device.** 5. **The device can then interpret the data and display the temperature and humidity readings.** 6. What are some possible sources of error when using the DHT11 sensor, and how can these errors be minimized?  * **Reading timing: The DHT11 sensor requires a specific timing sequence to read temperature and humidity data.** * **Temperature fluctuations: The DHT11 sensor measures temperature and humidity based on the thermal properties of its sensing element. Any sudden changes in temperature can cause thermal shock and affect the accuracy of the readings.** * **Interference from other electronics: The DHT11 sensor can be affected by electromagnetic interference (EMI) from other electronic devices in its vicinity.**  1. Can you think of any real-world applications where the DHT11 sensor might be used?   **This sensor could be used in food storage, medical applications, agriculture and how we used it, which was to measure the temperature and humidity.**   1. How could you modify the code used in the practice to display the temperature and humidity readings on an LCD screen instead of the serial monitor?   **#include <LiquidCrystal.h>**  **LiquidCrystal lcd(2, 3, 4, 5, 6, 7);**  **const int lm35Pin = A0;**  **const int humi = A1;**  **void setup() {**  **Serial.begin(9600);**  **lcd.begin(16, 2);**  **}**  **void loop() {**  **int lm35Value = analogRead(lm35Pin);**  **int humidityValue = analogRead(humi);**  **float humidity = (humidityValue / 876.0) \* 100.0;**  **float temperature = (lm35Value \* 5.0 / 1023.0) / 0.01;**  **lcd.clear();**  **lcd.setCursor(0,0);**  **lcd.print("Temperature: ");**  **lcd.print(String(temperature-50));**  **lcd.print(" C");**  **lcd.setCursor(0,1);**  **lcd.print("humidity: ");**  **lcd.print(String(humidity));**  **lcd.print(" %");**  **delay(1000);**  **}** | | | | | | | |
|  | | | | | | | |
| 1. **Results and conclusion** | | | | | | | |

**Fecha de realización:** 11-05-2023

*El alumno declara la originalidad de su reporte de práctica, que la elaboración del reporte y la práctica son de su autoría y que todo lo mencionado en el reporte es verdad.*