Steps to Set Up RTC DS1307 with Arduino

Hardware Required:

1. **Arduino Board** (Uno, Mega, etc.)
2. **DS1307 RTC Module**
3. **I2C LCD Display (Optional)**
4. **Jumper Wires**
5. **Battery (CR2032) for RTC module** (to keep time running when power is off)

Wiring:

DS1307 RTC	**Arduino**
VCC	5V
GND	GND
SDA	A4 (I2C Data)
SCL	A5 (I2C Clock)

Code for Testing RTC DS1307

This code reads the real-time from the DS1307 and displays it in the Serial Monitor.

Install Required Library:

Before running the code, install **RTClib** from Arduino Library Manager.

1. Open Arduino IDE
2. Go to **Sketch → Include Library → Manage Libraries**
3. Search for **RTClib**
4. Install **RTClib by Adafruit**

Arduino Code:

```
#include <Wire.h>

#include <RTClib.h>


RTC_DS1307 rtc;
```

```
const int buzzerPin = 7;  // Buzzer connected to Pin 7
const int alarmHour = 3;  // Set to match your test time
const int alarmMinute = 11; // Set to match your test time

void setup() {
  Serial.begin(9600);
  Wire.begin();

  pinMode(buzzerPin, OUTPUT);
  digitalWrite(buzzerPin, LOW); // Ensure buzzer is OFF at start

  if (!rtc.begin()) {
    Serial.println("RTC not found! Check wiring.");
    while (1);
  }

  if (!rtc.isrunning()) {
    Serial.println("RTC is NOT running!");
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  }
}

void loop() {
  DateTime now = rtc.now(); // Get current time from RTC

  // Display Time in Serial Monitor
  Serial.print("Current Time: ");
```

```
  Serial.print(now.hour());
  Serial.print(":");
  Serial.print(now.minute());
  Serial.print(":");
  Serial.print(now.second());
  Serial.println();

  // Check if it's time for the alarm
  if (now.hour() == alarmHour && now.minute() == alarmMinute && now.second() == 0) {
    Serial.println("ALARM! Buzzer ON!");
    tone(buzzerPin, 10000); // Generate sound at 1000 Hz
    delay(10000);  // Keep buzzer ON for 5 seconds
    noTone(buzzerPin);  // Turn OFF buzzer
    Serial.println("Buzzer OFF");
  }

  delay(1000); // Update every second
}
```