**Internet Clock using ESP32 and OLED Display**

In today's interconnected world, accurate timekeeping is essential for various applications and devices. From coordinating global financial transactions to ensuring synchronization in distributed systems, having a reliable and precise time reference is crucial. With the advancement of technology, it is now possible to create an Internet clock that can fetch and display accurate time from online sources. In this tutorial, we will explore the fascinating world of building an Internet clock using the ESP32 microcontroller and an OLED display.

The ESP32, a powerful and versatile microcontroller, provides the necessary capabilities for connecting to the internet and retrieving time data from online servers. Combined with an OLED display, which offers a crisp and clear visual output, we can create a compact and visually appealing clock that displays the accurate time information.

Through this tutorial, we will delve into the step-by-step process of building the Internet clock. We will cover the necessary hardware components, including the ESP32 microcontroller and the OLED display, as well as the required software libraries and tools for programming.

**Purpose of the system**

The purpose of the Internet clock system using ESP32 and OLED display is to provide an accurate and synchronized time reference by fetching time data from online servers. The system aims to offer a reliable and visually appealing solution for displaying the precise time, eliminating the need for manual adjustments and ensuring synchronization with global time standards.

With this system, users can benefit from the convenience of having an internet-connected clock that automatically retrieves accurate time information from online sources. It eliminates the reliance on traditional clocks that may require manual setting and can potentially drift in accuracy over time. The Internet clock offers a hassle-free and highly reliable alternative, especially in environments where precise timekeeping is critical, such as financial transactions, industrial automation, or distributed systems.

**Advantages of the system**

The Internet clock system using ESP32 and OLED display offers several advantages over traditional clocks and timekeeping methods. Here are some key advantages:

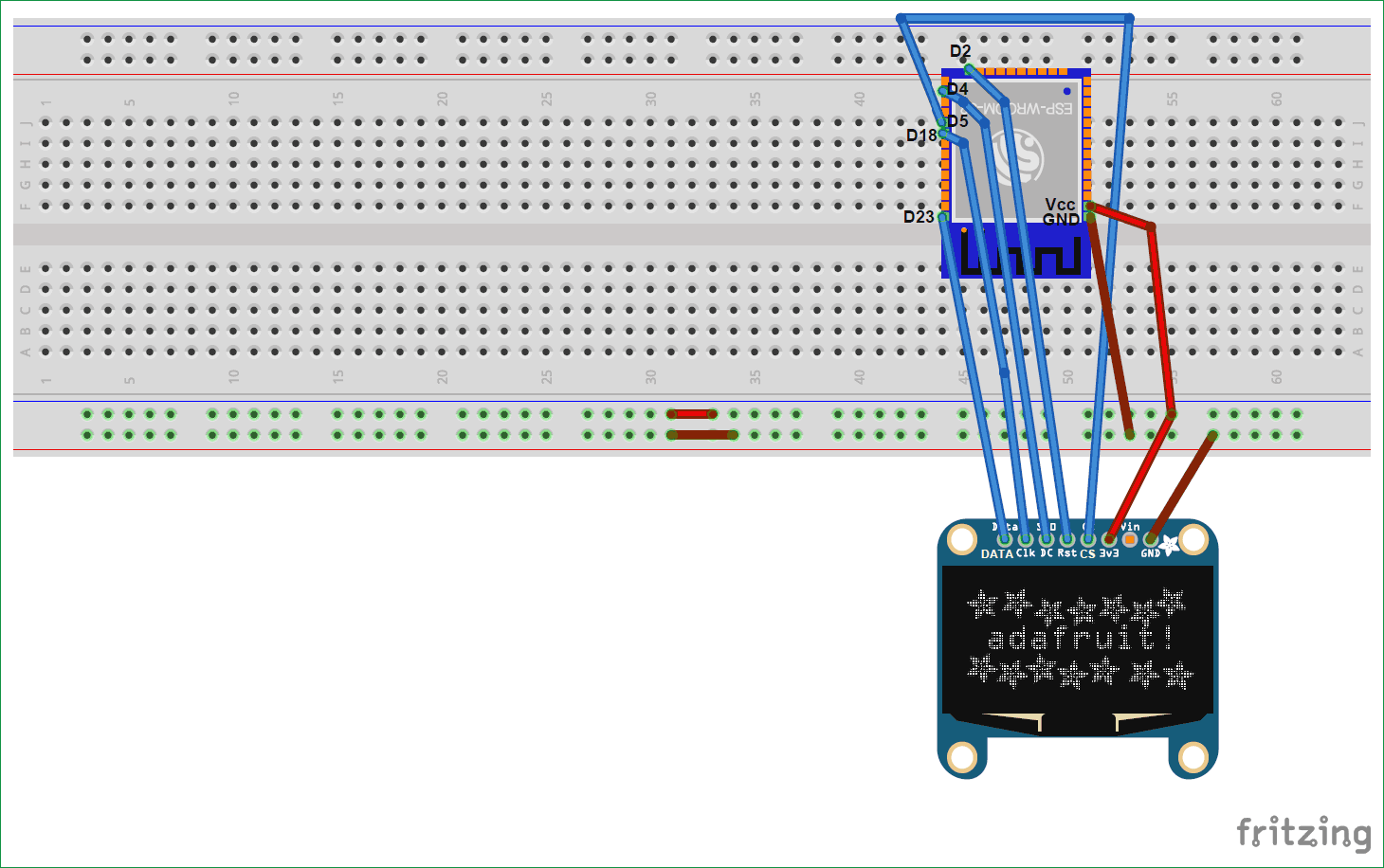
* Accuracy: The system fetches time data from online servers, which are synchronized with highly accurate atomic clocks. This ensures that the displayed time is precise and synchronized with global time standards, eliminating the need for manual adjustments.
* Synchronization: The Internet clock system allows for automatic synchronization with online time servers, ensuring that the displayed time remains consistent and in line with the most accurate time references available. This is particularly useful in environments where time synchronization is critical, such as in distributed systems or networked applications.
* Convenience: With the Internet clock system, there is no need for manual time setting or adjustments. The system automatically fetches the time information from online sources, providing a hassle-free and effortless timekeeping experience.
* Real-time Updates: The system can continuously update the displayed time in real-time, providing users with the most up-to-date and accurate time information. This is particularly beneficial in scenarios where time-critical decisions or actions need to be made.
* Versatility: The ESP32 microcontroller used in the system offers versatile connectivity options, allowing for easy integration with various networks and online time servers. This makes the Internet clock system adaptable to different environments and usage scenarios.

**Building a prototype**

**Components**

1. ESP32
2. Breadboard
3. Jumper wires(Male and female)
4. 128\*64 OLED display

**Circuit diagram**



We used SPI mode to connect our 128×64 OLED display Module (SSD1306) to ESP32. So, it will use 7 pins. Connections with ESP32 are given as:

|  |  |
| --- | --- |
| OLED Pins | Connected to ESP32 pins |
| 1. CS 2. DC 3. RES 4. SDA 5. SCK 6. VDD 7. GND | 1. D5 2. D4 3. D2 4. D23 5. D18 6. Vcc 7. GND |

**Code:**

#include <WiFi.h>

#include <SPI.h>

#include <Adafruit\_GFX.h>

#include <NTPClient.h>

#include <WiFiUdp.h>

#include <Adafruit\_SSD1306.h>

const char\* ssid     = "\*\*";

const char\* password = "\*\*";

#define NTP\_OFFSET  19800 // In seconds 

#define NTP\_INTERVAL 60 \* 1000    // In miliseconds

#define NTP\_ADDRESS  "1.asia.pool.ntp.org"

WiFiUDP ntpUDP;

NTPClient timeClient(ntpUDP, NTP\_ADDRESS, NTP\_OFFSET, NTP\_INTERVAL);

#define OLED\_MOSI  23

#define OLED\_CLK   18

#define OLED\_DC    4

#define OLED\_CS    5

#define OLED\_RESET 2

Adafruit\_SSD1306 display(OLED\_MOSI, OLED\_CLK, OLED\_DC, OLED\_RESET, OLED\_CS);

void setup()

{

display.begin();

Serial.begin(9600);

Serial.println();

Serial.println();

Serial.print("Connecting");

Serial.println(ssid);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED)

{

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.println("WiFi has been successfully connected.");

Serial.println("The IP address is: ");

Serial.println(WiFi.localIP());

timeClient.begin();

display.begin(SSD1306\_SWITCHCAPVCC);

display.clearDisplay();

display.setTextColor(WHITE);

display.setTextSize(2);

}

void loop()

{

timeClient.update();

String formattedTime = timeClient.getFormattedTime();

display.clearDisplay();

display.setTextSize(2);

display.setCursor(0, 0);

display.println(formattedTime);

display.display();   // Please write the buffer to the display

delay(10);

delay(100);

}

We have to define three variables in the programme for NTP in order to display the time from the Internet.

* NTP\_OFFSET, which is your country's time zone; for example, +5:30 for India. So in seconds, it will be 19800.
* NTP\_INTERVAL, which is the amount of time that NTP needs to update the time. There are 60–64 seconds.
* Your nation's NTP server is located at NTP\_ADDRESS. You can use "in.pool.ntp.org" for India.

**Working of the system**

The working of the Internet clock system using ESP32 and OLED display involves several steps to fetch and display accurate time information. Here is a detailed overview of the system's operation:

* Hardware Setup: Begin by connecting the ESP32, OLED display, and breadboard using jumper wires. Ensure proper connections between the ESP32's GPIO pins and the corresponding pins on the OLED display.
* Initialization: Import the necessary libraries and modules on the ESP32. This includes configuration of the OLED display, establishing Wi-Fi connectivity, and setting up the necessary time-related functions.
* Wi-Fi Connection: Connect the ESP32 to a Wi-Fi network for the internet connection. This allows the ESP32 to fetch time data from online servers.
* Time Synchronization: Use the ESP32's built-in functions or external libraries to synchronize the system's time with an online time server. This helps to make sure that the system's internal clock is accurate and up-to-date.
* Time Retrieval: Fetch the current time from the online time server using the ESP32's internet connectivity capabilities. This can be done through protocols such as Network Time Protocol (NTP) or Simple Network Time Protocol (SNTP).
* Display Update: Update the OLED display with the retrieved time information. Utilize the appropriate functions or libraries to convert the time data into a suitable format and display it on the OLED screen. This may include formatting the time in hours, minutes, and seconds, along with any additional details such as date or time zone.
* Continuous Update: Implement a loop mechanism to continuously fetch and update the time information from the online server. This ensures that the displayed time remains accurate and synchronized.
* User Interaction: Optionally, incorporate user interaction features such as button presses to adjust settings, switch between time zones, or display additional information on the OLED screen.
* By following these steps, the Internet clock system using ESP32 and OLED display can fetch the precise time from online servers and display it on the OLED screen. The system continuously updates the time display to ensure accuracy and synchronization with the global time reference.

**Conclusion**

In conclusion, the Internet clock system utilizing the ESP32 microcontroller and OLED display is a testament to the convergence of advanced technology and precise timekeeping. With its flawless and seamless integration of online connectivity and visual output, this innovative system offers an incredible solution for accurate time retrieval and display. By utilising the ESP32's robust capabilities and the high-resolution OLED display's visual capabilities, users are presented with a simple timekeeping experience that transcends conventional methods.

Through its ability to synchronize with online time servers and continuously update the displayed time, this system ensures temporal precision without any manual intervention. The harmonious interplay between the ESP32, OLED display, and Wi-Fi connectivity harmonizes seamlessly to bring forth an unparalleled timekeeping marvel.

Moreover, the Internet clock system embraces convenience, obliterating the need for cumbersome adjustments and ushering in a new era of effortless time management. The visual allure of the OLED display, complemented by the ESP32's impeccable performance, delivers a feast for the eyes, captivating users with its captivating and refined presentation of the time.