Adriann Liceralde
Dr. Brian McPherson
Carbon Science Research at the University of Utah
Wireless Sensor Network Project
Last Updated March 18, 2022

Assessment of the DS3231 Real-Time Clock

I. Introduction

Timestamps are crucial for data-logging applications. For example, in the *Wireless Sensor Network (WSN)* project, attaching a reliable timestamp to each data point is necessary for detecting and locating carbon sources. The local clocks within computers are accurate, especially when they regularly synchronize with an online clock. However, an external time-keeping module is necessary for systems detached from a local computer. The Sensor Nodes in *WSN* are deployed across a field and thus require an external clock.

The DS3231SN is a highly accurate, low-cost real-time clock (RTC)[1]. Each chip is typically sold for under $10, making it an attractive selection. In addition, the power consumption is minimal. However, temperature changes can affect a clock's accuracy. Therefore, the DS3231SN has an integrated temperature-compensated crystal oscillator that adjusts the clock accordingly.

II. Time Drift

According to the datasheet of the RTC, the chip has an accuracy of ±3.5ppm[1]. Inputting this accuracy into the following equation calculates the chip's maximum error per day:

$$error = \frac{accuracy\ (ppm)}{1000000\ (ppm)} * 86400 \frac{(seconds)}{(day)}$$

With the rated accuracy of ±3.5ppm, the RTC theoretically deviates from the set time of about ± 0.3 seconds per day. This error is comparable to the 0.10 seconds that a dedicated time website has[2]. For most time-keeping applications, these errors are reliable and acceptable. However, over time, the daily errors of a clock accumulate into an observable offset. For example, a clock that drifts 0.3 seconds per day becomes one second off from real-time three days later.

Frequent synchronizations prevent significant time drifts from occurring. In the case of websites and computers, the clocks synchronize with an online server, such as the Windows time server[3]. For the DS3231SN chip, the RTC must frequently sync with a master clock at regular intervals. This assessment aims to determine the rate at which the DS3231SN chip experiences time drifts. If the time drift rate is high, then synchronizations must be frequent for the RTC to remain reliable.

III. Observing the Time Drift

a. Equipment

The DS3231SN chip is operable by an Arduino microcontroller using the I^2C communication protocol. The I^2C address of the RTC is immutable; therefore, an Arduino board can only control one RTC module at a time. However, using an I^2C multiplexer module allows multiple I^2C devices with immutable addresses to communicate with one Arduino board.

The circuit in this assessment consists of an Arduino Nano, an I^2C multiplexer, and four RTC modules, as shown in Figures 1 and 2. Observing the time drifts of four RTC modules allows more information to be gathered. Theoretically, all modules should experience a similar rate of time drift. However, if one module drifts significantly faster than the others, it indicates that the DS3231SN modules vary in performance from copy to copy.

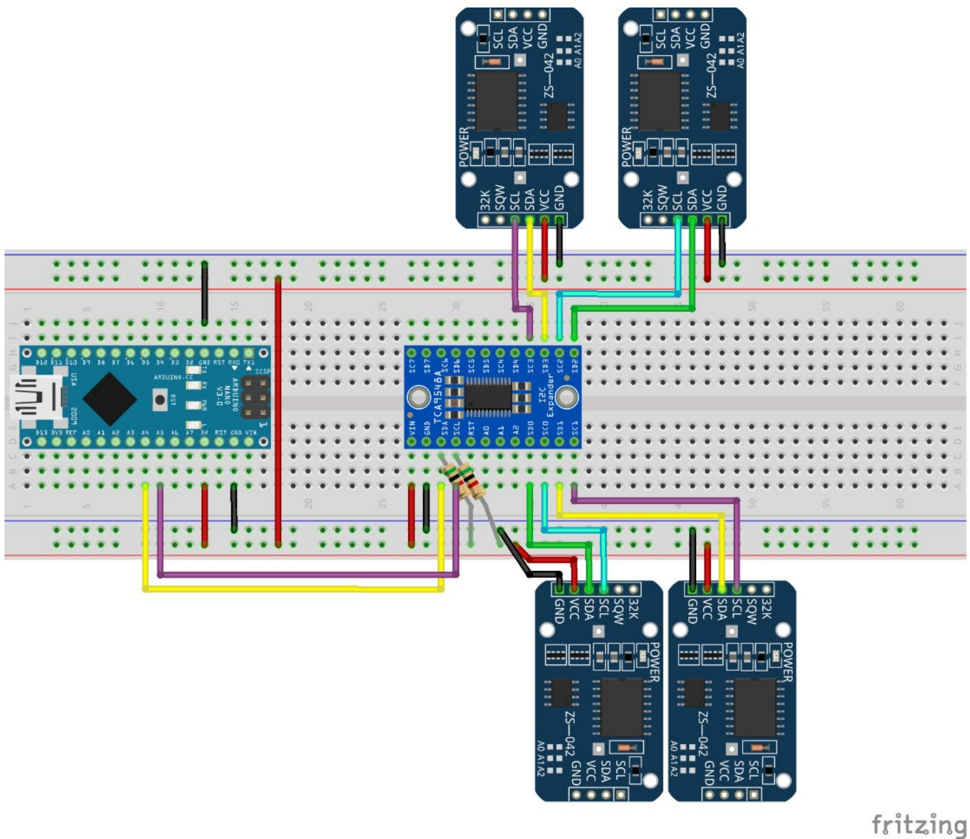

Figure 1. Breadboard diagram of the RTC Multiplexer.

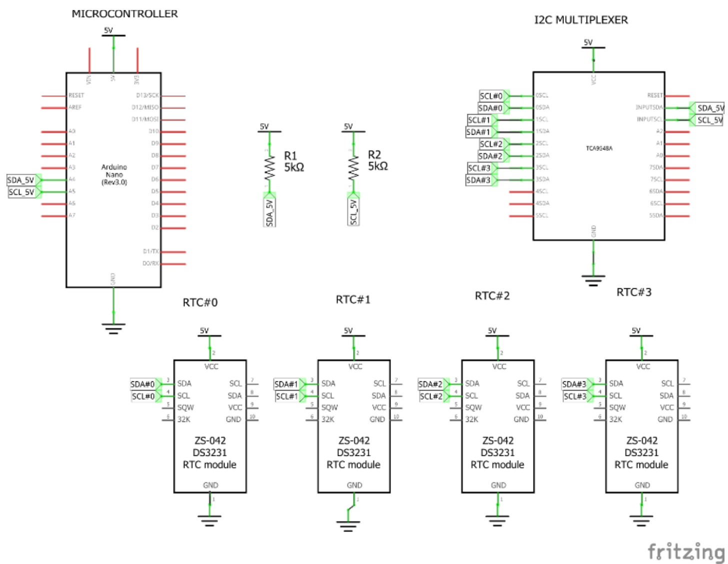

Figure 2. Circuit Diagram of the RTC Multiplexer.

b. Setup

The procedure for operating the RTC Multiplexer is as follows:

1. Assemble the multiplexer containing four identical RTC modules
2. Connect the Arduino to a computer
3. Upload the data-logging code to the Arduino board
4. The code automatically collects the timestamps every second from each RTC module and the timestamp given by the computer
5. Run the code for any desired duration

c. Results

The observation performed in this assessment spanned about two weeks. Afterward, the data was analyzed with Python and the Matplotlib library. The difference between the RTC timestamps and the local computer's time was calculated at each data point. Figure 3 shows the hourly averages of the time deviations.

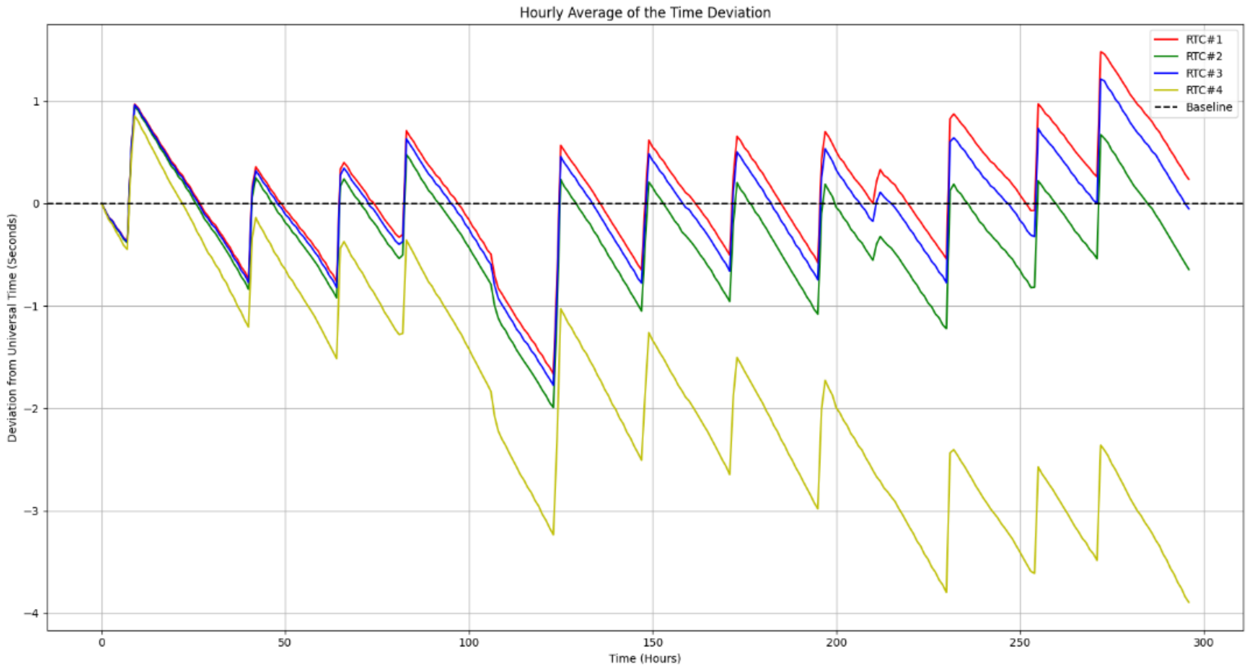

Figure 3. Hourly Average of the Time Deviations from the Baseline.

An estimate for the daily time drift is calculated from the data. The fastest negative deviation in a given day was -1.8 seconds, while the fastest positive deviation was only 0.7 seconds. Therefore, the time drift ranged from -1.8 to 0.7 seconds per day. On average, the daily drift was -0.06 seconds per day.

d. Analysis

The results from this assessment reveal helpful information about the DS3231SN chips. Below are key observations taken by analyzing the hourly averages of the time deviations:

1. The plots have a zig-zag behavior, likely due to the differences in clock precision. For example, the local computer timestamp gives seconds with 3 decimal places, while the DS3231SN gives no decimal places for its seconds. In addition, the graph represents hourly averages. So an average in one hour may be significantly different than the next hour's average, thereby creating a zig-zag. However, despite the zig-zag behavior, general trends can be seen from the graph.

2. RTCs #1, #2, and #3 generally oscillated around the baseline for most of the duration. However, a slight upward trend is noticeable after time t = 230 hours. At the end of the 2-week duration, these RTCs were slightly ahead of the baseline. This trend demonstrates how time drifts overtime can accumulate into a slight offset.

3. RTC #4 did not oscillate around the baseline. Instead, there was an early downward trend of time drifting. As a result, the time drift of #4 accumulated into a negative direction; overtime, the clock fell further behind than the baseline. This downward drift was so prominent that after 2 weeks, the clock was behind by about 4 seconds from the baseline, thus demonstrating a more significant effect of the accumulation of time drifts.

4. Most of the RTC modules showed extreme accuracy. For example, RTCs #1, #2, and #3 had a one-second offset from the baseline after 2-weeks of operation, thereby showing how well the DS3231SN can keep time. However, the only exception is RTC #4, which had a final offset of almost 4 seconds. Despite a more significant offset, a time drift of 4 seconds in two weeks is still reasonable. Thus, the DS3231SN chip has extreme accuracy, as marketed.

5. Although the DS3231SN chips perform well, individual chips may vary from each other. The variation of chips is demonstrated by how RTC #4 had a faster time drift rate than the other chips. In addition, it drifted in the opposite direction. Therefore, it should be expected that other DS3231SN chips may perform differently from those in this assessment.

IV. Conclusion

The theoretical maximum error given by the DS3231SN's datasheet is ± 0.3 seconds per day. However, the observed rate ranged from -1.8 to 0.7 seconds per day, while the average rate was -0.06 seconds per day, which corroborates the theoretical rate. Due to the slow time drift rate, the DS3231SN requires time synchronizations of only once per day or less. A daily synchronization ensures that the local RTC chip remains consistent with the universal time.

The Wireless Sensor Network consists of distributed sensor nodes, each with its own RTC chip. Each sensor node is wirelessly connected to a central hub that contains the master clock that sets the baseline for all the RTC chips in the network. Daily transmission of a time sync minimizes the effects of time drifts.

*Additional resources are contained in the Wireless Sensor Network repository[4] and the CSR Arduino Collection repository[5].

References and Useful Links

[1] RTC Datasheet
https://datasheets.maximintegrated.com/en/ds/DS3231.pdf

[2] Time Website
https://time.is/about

[3] Computer Time Synchronization
https://blog.codinghorror.com/keeping-time-on-the-pc/

[4] Wireless Sensor Network Repository
https://github.com/jkub6/WirelessSensorNetwork

[5] CSR Arduino Collection Repository
https://github.com/RiceAllDay22/CSR_Arduino_Collection

Contact

For any questions or assistance, email Adriann Liceralde at adriann8399@gmail.com