

UART Receiver Synchronization: Investigating the Maximum Tolerable Clock Frequency Deviation

Mitu Raj*

Department of Electronics and Communication, Electronics Research and Development Center of India - Institute of Technology, Center for Development of Advanced Computing, Vellayambalam, Trivandrum – 695010, Kerala, India; iammituraj@gmail.com

Abstract

Objectives: To analyse the maximum tolerable clock frequency deviation in UART receiver for accurate data reception and to derive mathematical expressions for the same. **Methods/Analysis:** General design techniques and synchronisation problems in UART receivers have been investigated and a mathematical analysis of the designs is done. Various parameters taken into consideration are baud rate, parity/no parity, oversampling ratio and phase difference between transmitter and receiver clocks. **Findings:** The maximum tolerable frequency deviation, $\Delta_{\max}$ improves with increasing oversampling ratio at the UART Receiver side. It is also found that the $\Delta_{\max}$ is independent of the baud rate of operation. **Improvements:** Improved expressions may be derived by analyzing the effects of clock jitter, rise and fall times of the signals.

Keywords: Sampling, Serial Communication, Synchronization, UART

1. Introduction

Serial communication is the most commonly used method of data communication in embedded systems. In serial communication, data is sent or received serially, one bit at a time. UART or Universal Asynchronous Receiver Transmitter is among the most popular serial protocols

existing now. UART communication is typically full duplex in nature; i.e. both transmission and reception are possible at the same time¹. What makes UART different from other serial protocols, is its asynchronous nature. It will be discussed in the next section. An example of UART communication between two devices is shown in Figure 1.

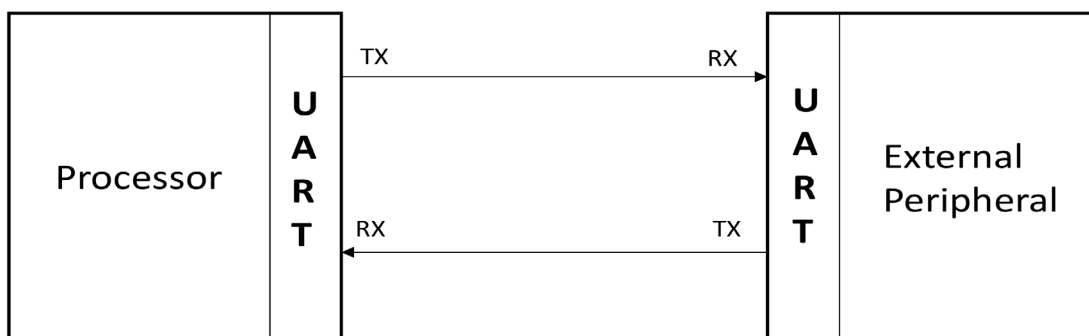


Figure 1. UART communication between two devices.

*Author for correspondence

In the above figure, TX is the UART port for serial data transmission and RX is the port for serial data reception.

2. Block Diagram and Data Frame of UART

Basic block diagram of UART is shown in Figure 2. In the figure, 'clock_in' is the input clock to the baud generator. The baud generator generates two clock from clock_in: clock_tx and clock_rx, for the transmitter and receiver parts respectively. The frequencies of the generated clocks depend on the defined baud rate for transmission/reception.

Baud rate signifies the no. of bits transmitted/received per second. UART Transmitter just converts the parallel data input into streams of serial data at the output at the defined baud rate. UART Receiver converts serial data input into parallel data output. Commonly used baud rates in UART are 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600 and 115200 bps or bits per second. The data stream transmitted or received serially by UART, is composed of frames. Each data frame contains a start bit (logic low or '0'), 8 data bits and a stop bit (logic high or '1'). Typical data frame of a UART looks like in Figure 3.

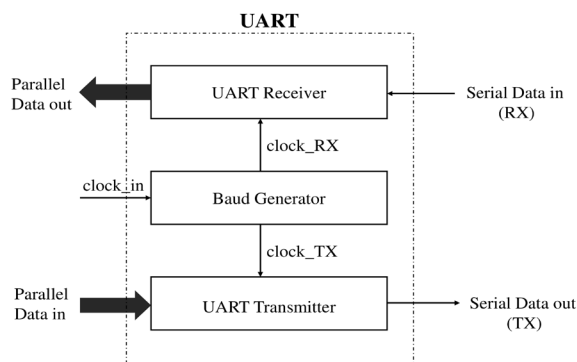


Figure 2. Basic block diagram of UART.

Start Bit	D0 (LSB)	D1	D2	D3	D4	D5	D6	D7	Parity Bit (optional)	Stop Bit
-----------	----------	----	----	----	----	----	----	----	-----------------------	----------

Figure 3. Typical data frame of UART.

3. UART Synchronization

UART communication is asynchronous; i.e. its transmitter and receiver operate in two different clocks, which

have no definite phase relationship between them. So UART Receiver should have some synchronization mechanism to synchronize itself to the incoming data stream and sample the data accurately. Consider a scenario in which an external peripheral is transmitting data serially at 9600 bps. To receive this data, the UART Receiver's clock should also sample the data at 9600 Hz. But problem arises, if the transmitted data have glitches at some instants. Glitches may be sampled as data by the receiver Figure 4. In the figure, second data bit is actually '0'. But it is wrongly sampled as '1' due to the presence of glitch at the instant of sampling.

To overcome this issue, oversampling technique is employed at the receiver end². The receiver's clock is thus oversampled to a high multiple of the transmitter's clock frequency Figure 5. Typically, an oversampling ratio of 8 or 16 is used in UART Receivers³⁻⁵. So in the above scenario, the UART Receiver's clock has to sample the data at 76800 bps (when oversampling ratio = 8x) or 153600 bps (when oversampling ratio = 16x). The incoming serial data are then determined as '0's or '1's by averaging the samples.

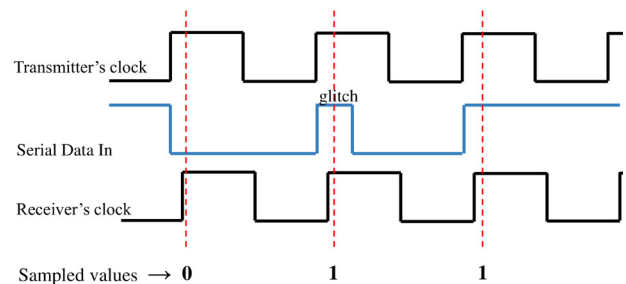


Figure 4. Sampling of glitch as data.

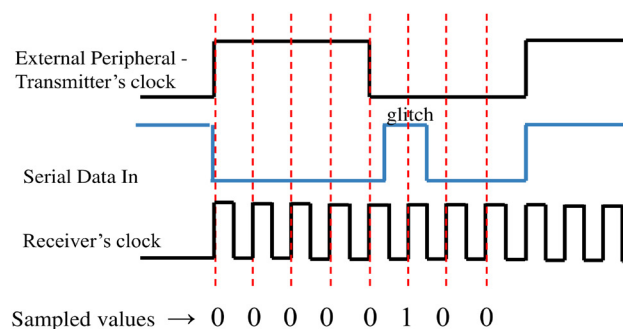


Figure 5. Oversampling at UART receiver.

3.1 Averaging the Samples

UART Receiver determines the logic level of the incoming data by sampling it and averaging the obtained samples. Consider the Figure 5, the sampling clock of the UART Receiver has an oversampling ratio = 8 times the baud rate (or External peripheral's transmitter clock frequency). The serial data are sampled every clock cycle and the samples are obtained as 1,1,1,1,0,1,1,1. They are then averaged and thus the received bit is determined as '1' by the UART Receiver.

4. Maximum Tolerable Frequency Deviation

As explained in the previous section, oversampling nullifies the effect of glitch in the data reception. But what if the UART receiver is not clocked exactly at the required oversampled frequency, and what if its period deviates by a small error value Δ ? This delta error accumulates over every clock cycle as shown in Figure 6. Eventually the receiver will lose synchronization and starts sampling wrong data. The delta error, Δ can be negative or positive.

To negate this undesirable behavior, UART Receiver is usually designed to re-synchronize itself after each data frame, using the start and stop bits ⁶. This limits the error accumulation to only 9 bits (if no parity) or 9 transmission clock cycles. This error accumulation puts a limit on the maximum tolerable deviation ($\Delta_{\max}$) of UART Receiver's clock frequency, from the required sampling rate.

4.1 Deriving Expressions for $\Delta_{\max}$ for Different Oversampling Ratios

Let F_B be the baud rate of transmission from external peripheral and T_B be the corresponding baud period. Let F be the required clock frequency at UART Receiver and T be the corresponding time period of the sampling clock. Assume that, the receiver samples the data at an oversampling ratio S .

For errorless reception,

$$F = S \times F_B$$

$$\Rightarrow T = T_B / S$$

Let the actual sampling period be,

$$T_s = T + \Delta$$

As explained before, this delta error will accumulate over every clock cycle. Now consider the following two cases:

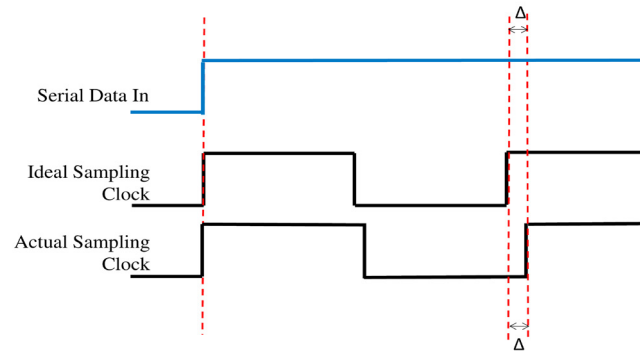


Figure 6. Delta difference in period between ideal and actual sampling clocks.

4.1.1 When Oversampling Ratio, $S = 8$

UART Receiver samples every received bit eight times. Once it samples a '0' for the first time, it is interpreted as the beginning of the start bit of the incoming frame. The UART Receiver then samples the subsequent data bits. To sample each received bit correctly, the total error accumulation up to that instant, should not exceed more than (or equal to) the time period of half the no. of samples/data bit. i.e. four samples, in this case. If it exceeds this limit, it is possible that the interpreted value of the received bit, after averaging the samples, is wrong. As explained in the previous section, the UART Receiver resynchronizes itself after every data frame. Therefore, the error accumulation has to be tolerated for 9 transmission clock cycles (i.e. 72 samples), till the sampling of stop bit (if no parity bit), to accurately interpret all the bits of the data frame. Hence, for errorless reception,

$$72 \times \Delta \leq 3T_s \quad (1)$$

$$\Rightarrow \Delta \leq 3T_s / 72$$

$$\Rightarrow \Delta \leq \frac{1}{24}(T + \Delta)$$

$$\Rightarrow \Delta / T \leq \frac{1}{23}$$

$$\text{i.e., } \Delta_{\max} = \pm 4.35\%$$

4.1.2 When Oversampling Ratio, $S = 16$

UART Receiver samples every received bit sixteen times. To sample each received bit correctly, the total error accumulation up to that instant, should not exceed more than (or equal to) the time period of half the no. of samples/

data bit. i.e., eight samples, in this case. So Equation (1) in this case, becomes:

$$144 \times \Delta \leq 7T_s$$

where $T_s = T_1/16$. Following the previous steps, it can be derived that:

$$\Delta_{\max} = \pm 5.11\%$$

Hence, it is observed that $\Delta_{\max}$ improves with increase in oversampling ratio.

4.1.3 General Expression and Other Dependencies

Studying the above results, a general expression can be derived for $\Delta_{\max}$ as:

$$9 \times S \times \Delta \leq \left(\frac{S}{2} - 1 \right) \times T_s \quad (2)$$

Finally, we obtain the expression for $\Delta_{\max}$ as:

$$\Delta_{\max} \leq \frac{(S-2)}{(17S+2)} \quad (3)$$

The Equation (2) can be modified to apply for UART data frames with parity bit, as follows:

$$10 \times S \times \Delta \leq \left(\frac{S}{2} - 1 \right) \times T_s$$

$$i.e., \Delta_{\max} = \frac{(S-2)}{(19S+2)} \quad (4)$$

It may also be noted that the falling edge of start bit and the edge of the first sample do not necessarily coincide, like shown in Figure 7. There may be a phase difference of ϕ between the edges, which causes the UART Receiver to miss at most one sample. Therefore, the maximum phase difference, $\phi_{\max} = T_s$. The Equation (2) may now be modified as:

$$(9 \times S \times \Delta) + \Phi_{\max} \leq \left(\frac{S}{2} - 1 \right) \times T_s$$

Finally, we obtain the expression for $\Delta_{\max}$ as:

$$i.e., \Delta_{\max} = \frac{(S-4)}{(17S+4)} \quad (5)$$

5. Results and Discussions

From the derived expressions, it may be observed that the maximum tolerable clock frequency deviation in UART

Receiver, $\Delta_{\max}$ improves with increasing oversampling ratio. Increase in $\Delta_{\max}$ means increase in data reception accuracy. The observation is graphically represented in Figure 8.

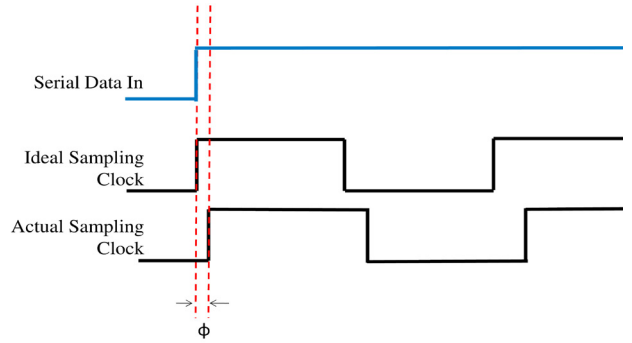


Figure 7. Phase difference between ideal and actual sampling clocks.

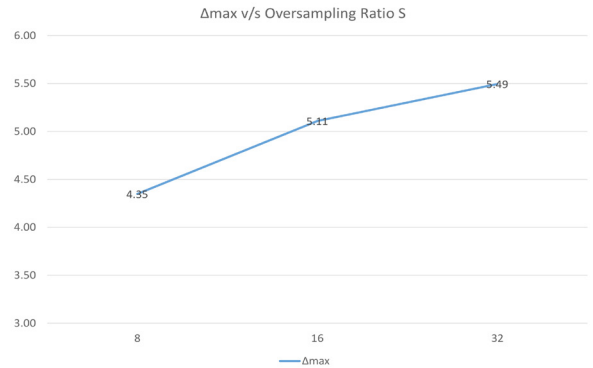


Figure 8. Variation of $\Delta_{\max}$ with oversampling ratio.

As the oversampling ratio increases above 32, slope of the graph is observed to be decreasing. The curve eventually flattens and $\Delta_{\max}$ almost becomes a constant. This limiting value of $\Delta_{\max}$ or the best possible value of $\Delta_{\max}$ can be calculated from equation (3) as:

$$\begin{aligned} & \lim_{S \rightarrow \infty} \left(\frac{S-2}{17S+2} \right) \\ &= \lim_{S \rightarrow \infty} \left(\frac{1 - \frac{2}{S}}{17 + \frac{2}{S}} \right) \\ &= \frac{1}{17} = \pm 5.88\% \end{aligned}$$

Comparing equations (3) and (4), it is clear that $\Delta_{\max}$ degrades when parity bit is included in the data frame. The $\Delta_{\max}$ is also found to be dependent on the phase difference, ϕ . From equations (3) and (5), it can be calculated that $\Delta_{\max}$ is always lesser in the latter case, for same values of S . For $S = 8$, $\Delta_{\max}$ is only $\pm 2.85\%$ when $\phi = \phi_{\max}$. At the same time, $\Delta_{\max}$ is $\pm 4.35\%$ when the phase difference is assumed to be zero. Hence, it infers that ϕ further degrades the data reception accuracy. It is also interesting to note that $\Delta_{\max}$ is independent of the baud rate of operation. The UART data reception can be concluded to be error prone, as long as it complies with the bounds put forth by the derived mathematical expressions.

6. Conclusion

The work investigates various synchronization problems that occur in UART Receiver and various design factors that affect the synchronization. There exists a maximum tolerable clock frequency deviation, $\Delta_{\max}$ at the receiver end, for accurate data reception. Various mathematical relations regarding the same have been derived, and it is found that $\Delta_{\max}$ is independent of the baud rate of the communication. It is also observed that $\Delta_{\max}$ increases with oversampling ratio. Thus it may be concluded that increasing oversampling ratio, improves the accuracy of data reception. Adding parity bit to the data frame, further

degrades $\Delta_{\max}$ and the accuracy of reception. Incorporating the effects of clock jitter, rise and fall times of signals may result in an improved mathematical expression.

7. References

1. Nanda U, Pattnaik SK. Universal Asynchronous Receiver and Transmitter (UART). Proceedings of the 3rd International Conference on Advanced Computing and Communication Systems; 2016 Jan. p. 1–5. Crossref
2. Ansari HK, Farooqi AS. Design of high speed UART for Programming FPGA. International Journal of Engineering and Computer Science. 2012 Oct; 1(1):28–36.
3. Bhadra D, Vij SV, Stevens SK. A low power UART design based on asynchronous techniques. IEEE 56th International Midwest Symposium on Circuits and Systems (MWSCAS); 2013 Aug. p. 21–4. Crossref
4. Dhanadravye A, Thorat S. A review on implementation of UART using different techniques. International Journal of Computer Science and Information Technologies. 2014 Jan; 5(1):394–6.
5. Laddha N, Thakare P. A review on serial communication by UART. International Journal of Advanced Research in Computer Science and Software Engineering. 2013 Jan; 3(1):366–9.
6. Agrawal RK, Mishra RV. The design of high speed UART. Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT 2013); 2013 Apr. p. 388–90. Crossref