Application Report

I²C Dataline Handoff Delay

Duy Nguyen, Jared Becker *Transceiver Interface*

ABSTRACT

This document discusses the SDA handoff delay during the eighth and ninth clock cycles of I²C transactions when the open drain drivers switch control of the bus. The TCA9517 device, an I²C static voltage offset buffer, is also examined for its propagation delay which can result in the SDA handoff delay.

Table of Contents

Trademarks

All trademarks are the property of their respective owners.

1 Introduction

I²C is a standard method of communication between multiple ICs within a system. It leverages a bus architecture with a controller device and either a single target or multiple target devices (see Figure 1-1). In some circumstances, the controller device and the target device may use two different voltage levels for I²C communication. In that case, the TCA9517 may be used to properly translate the voltage levels (see Figure 1-2). The TCA9517 is a buffered level-translator, meaning the device not only shifts the voltage levels to the needs of each device, but it also re-drives the signals in both directions. This buffer aspect of the TCA9517 can introduce propagation delays into the communication, which can result in visible SDA handoffs when recorded on an oscilloscope. This article explains what to expect on the SDA lines when the TCA9517 is used and if the result of the propagation delays from buffers like the TCA9517 are something to be concerned about.

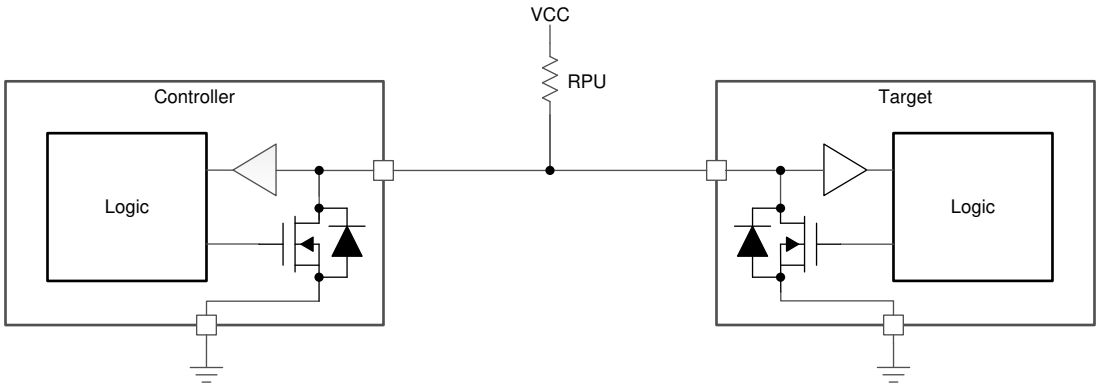

Figure 1-1. General I²C Diagram

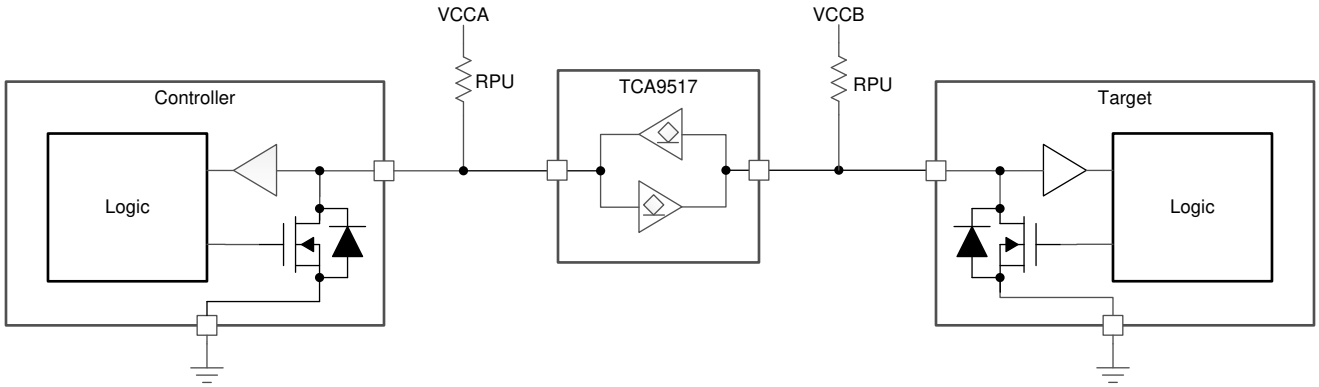

Figure 1-2. General I²C Diagram With TCA9517

2 SDA Signal During Acknowledge

During an I²C transaction, after the controller sends out the I²C address to the target device, the target should respond with an Acknowledge (ACK) to let the controller know the target recognizes its address and to signal to the controller to continue the transaction. This document focuses specifically on the ACK response from the target. Figure 2-1 shows the beginning of I²C communication between an I²C controller and I²C target in which the controller sends the address and the target responds with an ACK.

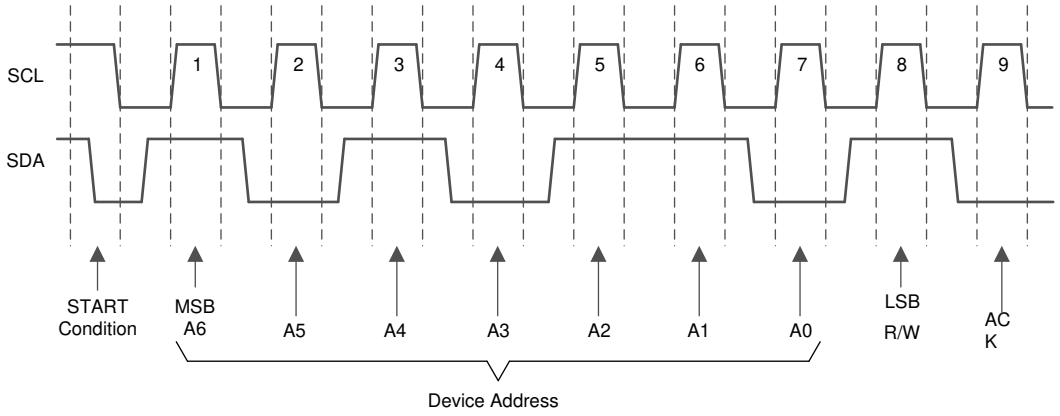

Figure 2-1. General I²C Timing

On the falling edge of the eighth clock cycle, the target takes control of the bus to signal to the controller that it has Acknowledged its address was called. Figure 2-2 shows the target driving the SDA line low after a short delay on the falling edge of the eighth clock pulse.

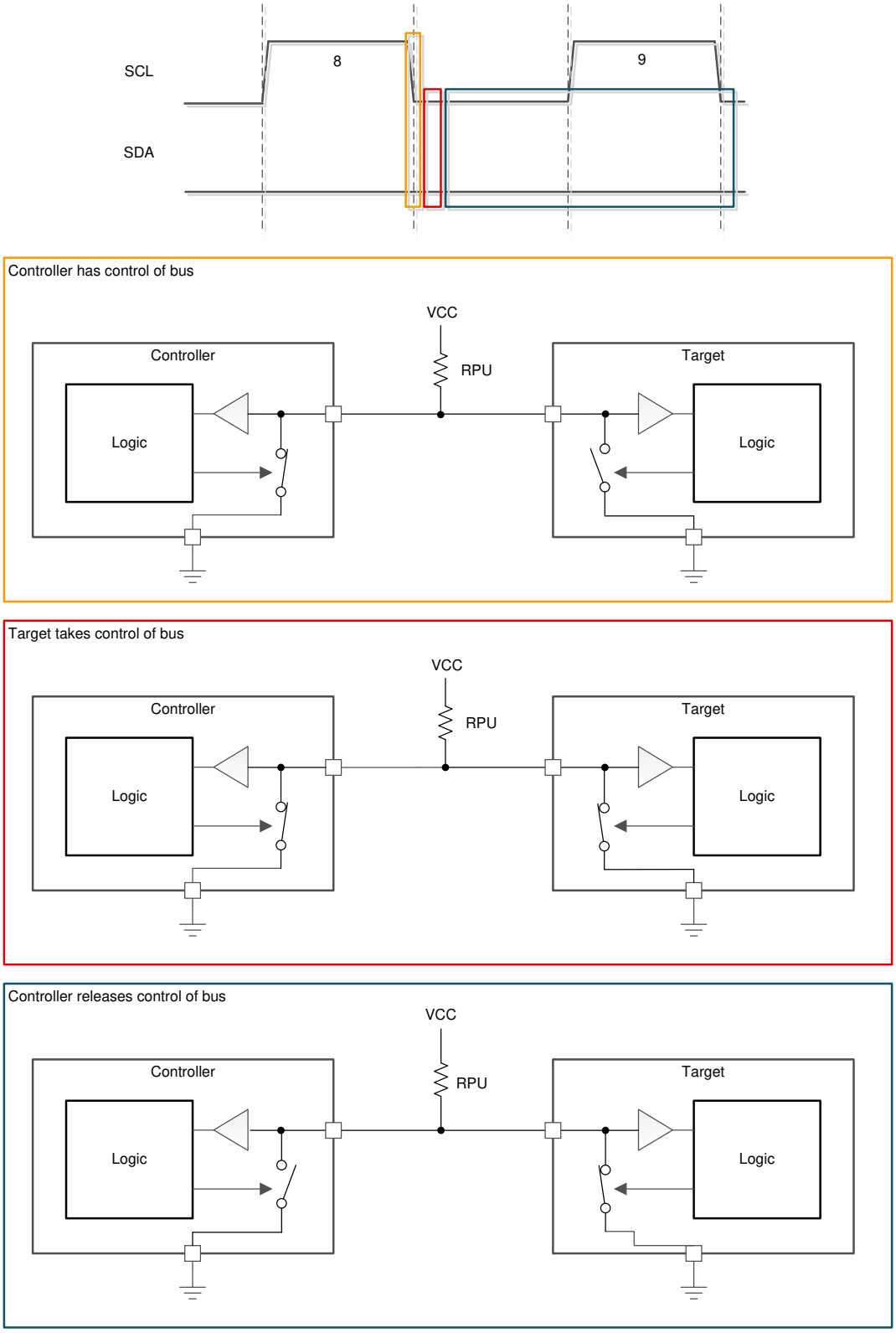

Figure 2-2. I²C SDA Handoff With Overlap

The yellow box shows illustrates the controller driving the SCL and SDA line low during a write transaction. The red box shows that there is a short period of time where both the controller and target have control of the bus after the eighth clock pulse, essentially an overlap of SDA being driven low by both controller and target. The purple box shows that the target has total control of the bus well before the start of the ninth clock pulse and through the duration of the ninth clock pulse. In this example, the open drain drivers of both the controller and the target overlapped and drove the SDA line low during the red box. This kind of example is common and occurs when the controller releases the SDA line after the target device drives the SDA line low.

3 SDA Signal During Acknowledge With Propagation Delay (TCA9517 A to B)

When the TCA9517 is added to the system for level translation or for redriving purposes, it introduces a propagation delay due to the integrated buffer design. Figure 3-1 shows an exaggerated timing diagram for the SCL and SDA lines with propagation delay added to the timing.

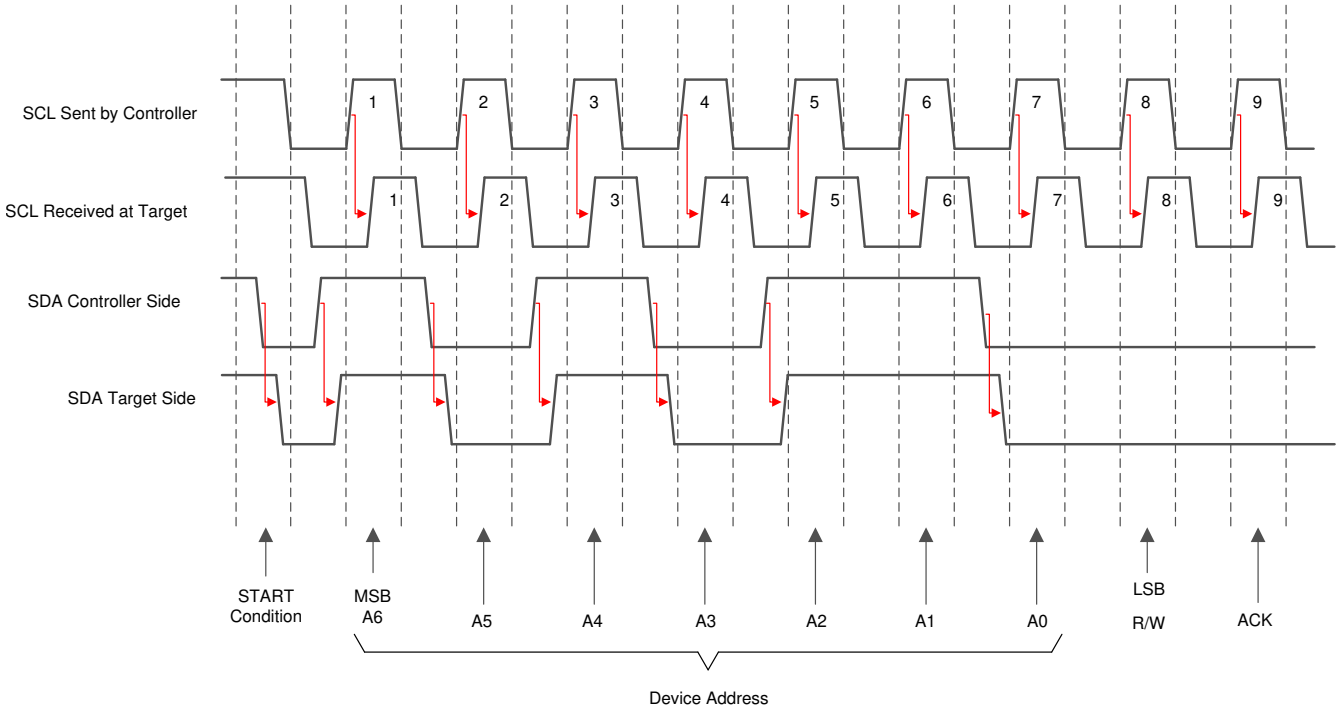

Figure 3-1. I²C Timing With Propagation Delays From I²C Buffer

From a data integrity standpoint, the signal delay does not impact the ability for the controller and target to communicate properly because the delay is equal for both the SCL and SDA lines. From a timing perspective, the SDA line remains stable for the entire portion of the SCL high period for the target device.

The propagation delay does impact the timing for the target device to take control of the bus during the ACK portion of the signaling with respect to the controller. Figure 3-2 shows a zoomed-in illustration of what occurs after the falling edge of the eighth clock pulse. After the falling edge of the eighth clock pulse, the open-drain driver of the controller needs to tri-state to give up control of the SDA line so that the target can take control of the SDA signal to ACK to the controller. However, there is a delay before the target receives the message to take control of the bus.

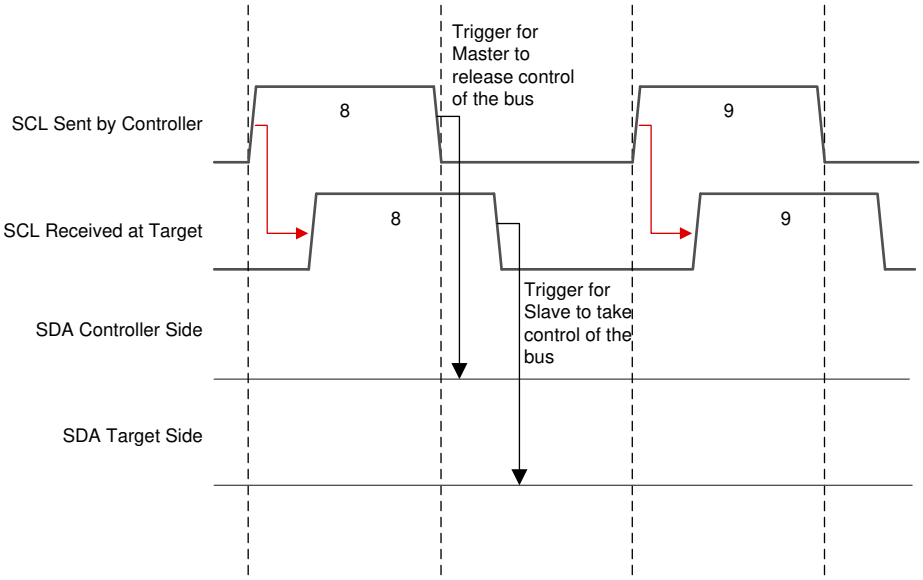

Figure 3-2. Eighth to Ninth Clock Cycle Zoomed in With Propagation Delay From Buffer

This delay in the SCL line being received by the target may result in a small window where the controller has released control of SDA, but the target has not yet taken control of the SDA signal yet because it has not seen the SCL signal go low. Figure 3-3 shows how this circumstance occurs in the system.

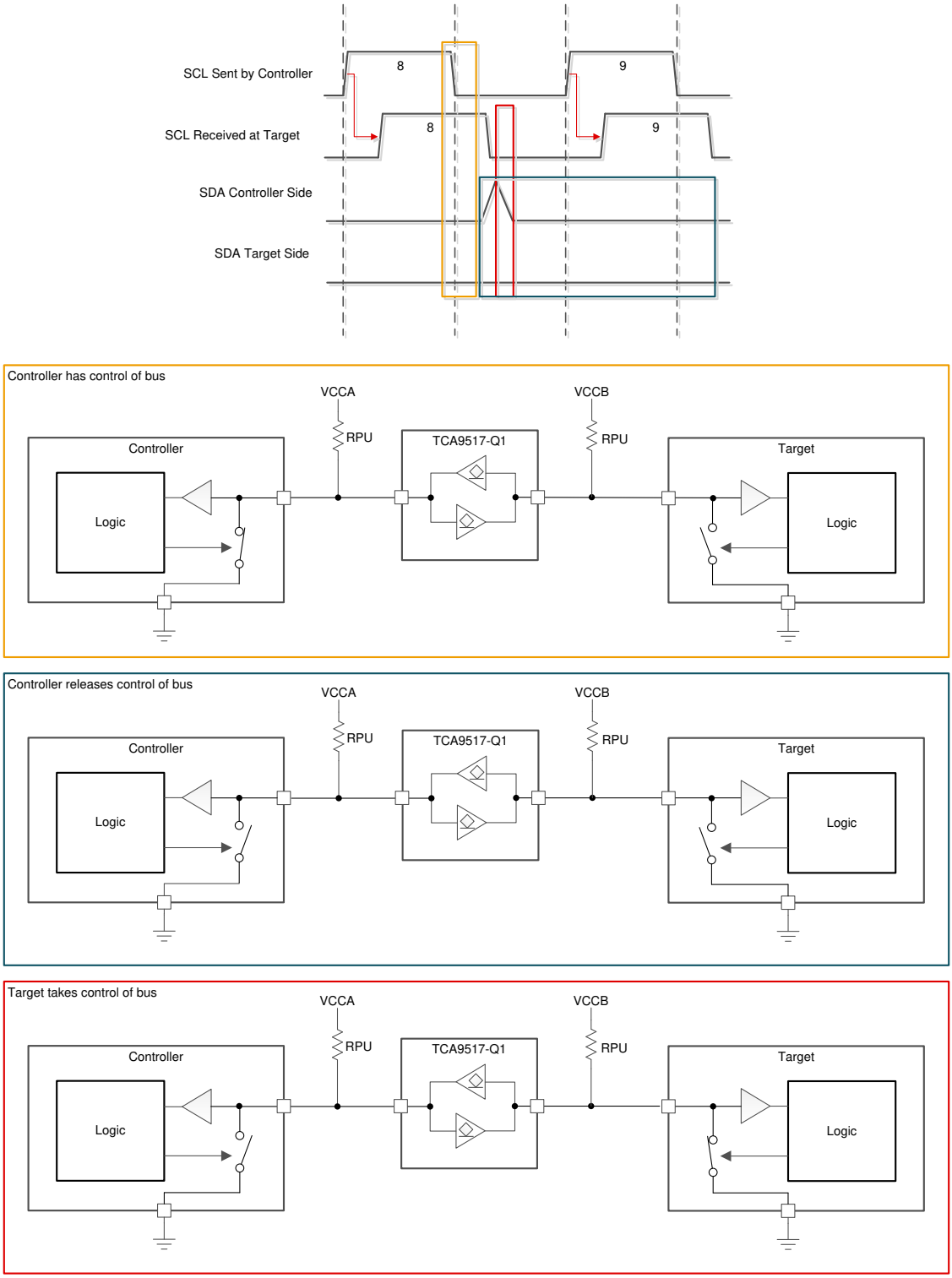

Figure 3-3. SDA Handoff Delay Illustration

During the window where the controller has released control of the bus and the target has not taken control of the bus, the SDA line on the side of the controller begins to rise. This occurs on the A side of the TCA9517, since there are no other external open-drain drivers driving the SDA line low. Before the rising SDA signal is able to traverse through the TCA9517 from the A side to the B side, the SCL signal has already reached the target device to signal for the target to take control of the bus. At this point, the target is able to take control of SDA before the SDA line starts to rise on the target side (the B side of the TCA9517 in this example).

3.1 Example Waveform in Actual System

To illustrate this effect in a real I²C transaction, Figure 3-4 shows a test setup of two devices communicating over the I²C bus with the TCA9517 included for level translation.

The controller side is connected to the A side of the TCA9517 and the target is connected to the B side of the TCA9517.

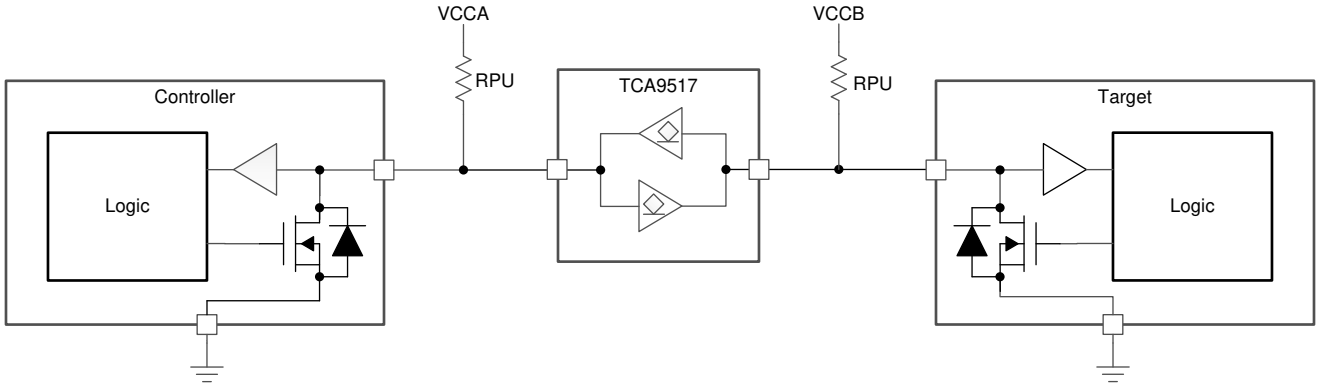

Figure 3-4. Example Setup

Figure 3-5 shows a screen capture of the SCL and SDA lines on both sides of the TCA9517. Looking at the region after the eighth clock pulse, a very short spike is visible on the SDA line of the controller.

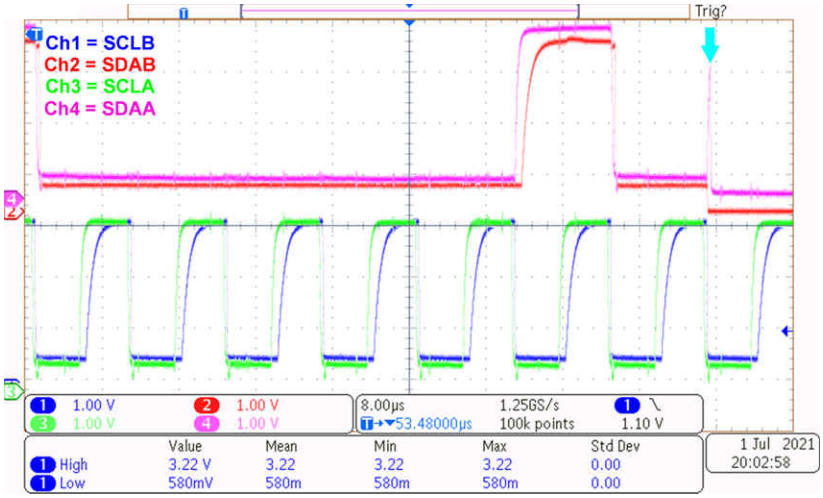

Figure 3-5. SDA Handoff Delay With Nine Clock Pulses

4 System Impact

Although adding the TCA9517 to the system can create a circumstance in which the signals generated by the controller are delayed by the TCA9517, there is no impact to the signal integrity of the communication. The *I²C-bus specification and user manual* shows in Section 3.1.3, Figure 4 that the data line must be stable when the SCL line is high, but is allowed to change states when the SCL line is low. The time in which the delay due to the TCA9517 occurs after the SCL line goes low and only happens while the SCL is low.

The I²C-bus specification does not specify the Acknowledgment timing for the controller and target devices. It does not state when the controller must release the bus for Acknowledgment and when the target must take control of the SDA line after the controller releases. However, it does specify a data set-up time requirement, which is found in Table 10 of the I²C standard. Looking at the requirement, each different operating mode requires a specific set-up time for the SDA line to be stable before the SCL line is at 30% of its rising voltage. Therefore, as long as the target device has control of the SDA line adequately before SCL line rises to 30% of its maximum voltage based upon the set-up time specification, the target is able to send a proper Acknowledge to the controller. Any voltage changes on the SDA line before the set-up time are simply ignored.

5 SDA Handoff Delay After the Acknowledge

The same SDA handoff delay signal that is seen on the eighth falling clock pulse can also happen at the ninth falling clock pulse. This can occur when the I²C target device releases the SDA line after having control of it during an ACK. If the I²C controller does not regain control of the SDA line quick enough, then the SDA signal can be seen rising before falling again which could look like an SDA pulse. This is illustrated in Figure 5-1. This is also acceptable from an I²C-bus standard perspective as long as the clock line is low and the set up time of the data line is still being met.

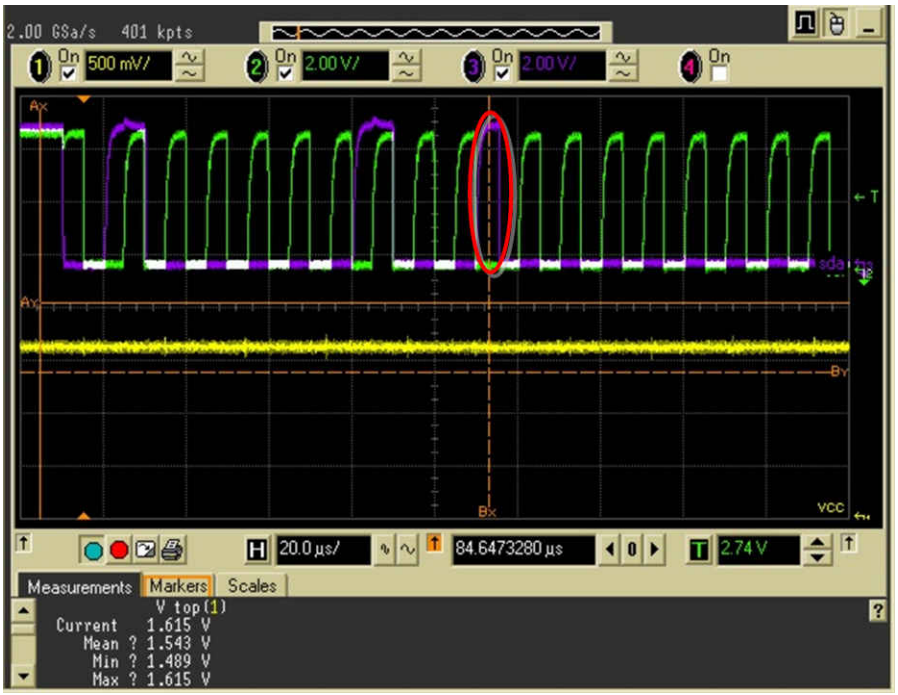

Figure 5-1. SDA Handoff Delayed Pulldown After Ninth Clock Pulse

6 Concern With Rise Time Accelerators

Rise time accelerators are included into specialized I²C translators or buffers to help drive heavy capacitive loaded I²C buses. This allows the external pullup resistors to be large to lower voltage output lows (VoLs). If the rise time accelerator circuit is designed to trigger upon only a voltage threshold level, the SDA hand-off delay can accidentally trigger the rise time accelerators to turn on and cause contention between the high-side driver of the rise time accelerator and the I²C device trying to pull the SDA line low. The consequence of this could cause damage to the devices under contention. So for this reason, the SDA lines on an I²C bus should be monitored to see if the SDA handoff delay occurs. If the SDA handoff delay occurs then the I²C system should be thoroughly tested across temperature, loading conditions, and boards whenever devices with rise time accelerators must be included on the bus where handoff delays are seen.

7 Conclusion

By adding a device with buffers to the I²C communication path, propagation delay is added to the signal transmission. The propagation delay can result in a small window of time where the controller has released the bus to allow for target to Acknowledgment, while the target has not yet taken control of the bus to send the Acknowledge. The impact of this is potentially a voltage spike on the SDA line of the controller, which is non-consequential to a traditional system. The consequence of this is impactful if there is a rise time accelerator on the I²C bus which may cause potential damage to the devices driving the bus if the rise time accelerators were to turn on due to the voltage spike during this window.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265