Why is an I²C Buffer Necessary?

- Expand the versatile I²C Bus
- Drive long cables or large PCBs
- Voltage Level Translation

Abstract:
As new applications emerge for the popular I²C bus components the use of application specific signal buffers are necessary. These simple devices are often overlooked yet deliver a more robust design; offering strong drive for long buses, segmentation for fault finding, and easy Voltage Level Translation.

Introduction

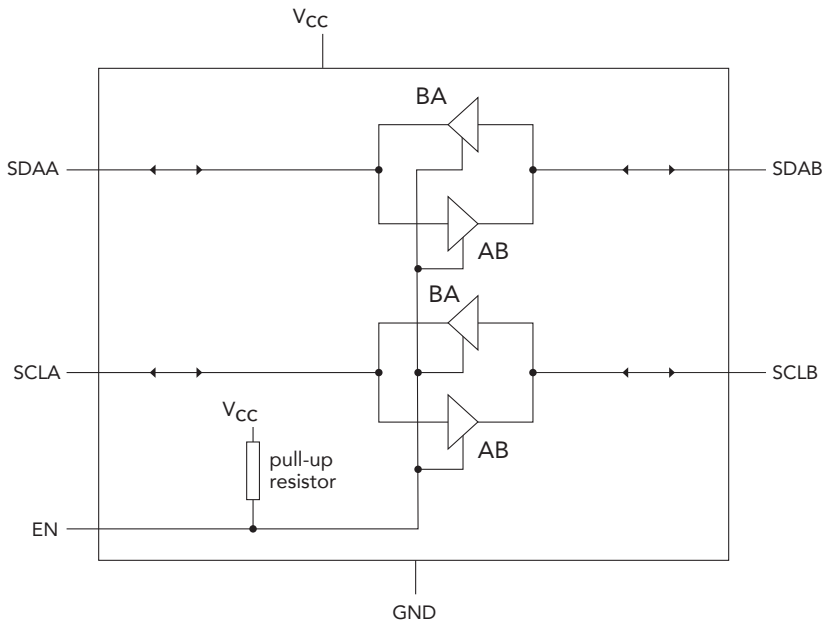
The I²C Bus simplifies communication between devices. The I²C bus is a true multi-master, multi-slave bus for serial communications.

The bus employs two bidirectional lines: Serial Clock and Serial Data. The initial implementation, Standard Mode, has a maximum speed of 100kHz. Speed improvements include Fast Mode at 400kHz and High Speed mode at 3.4MHz. Two newer improvements are Fast Mode Plus at 1 MHz, and Ultra Fast Mode at 5MHz. Newer devices are capable of the faster speeds and require lower supply voltages, but many legacy components are still in use. Faster speeds limit allowable capacitance, which can limit the number of devices allowed.

I²C Buffers

Incorporating I²C buffers into new designs will limit or negate the problems caused by higher device counts, extended bus lengths, and the different voltage and speed requirements when legacy components are mixed with newer components. NXP's no-offset buffer design avoids the inherent compromises of static and incremental offset buffers or drive current amplifiers. Multiple buffers in a star (parallel) configuration can segment a bus, and serial configurations can extend the I²C bus or drive a short cable between cards.

Figure 1. Functional Diagram of an I²C Buffer



Functional Description

A single buffer is constructed using two non-inverting buffer amplifiers connected in an opposing parallel configuration. The output of each amplifier is connected to the input of the other. As shown in Figure 1, each integrated circuit has two channels, one for the serial clock and one for the serial data.

The control logic within the integrated circuit determines which side of the bus has control and is active. When the A Side is active, the AB Buffer is enabled and the BA Buffer is disabled to prevent either amplifier from driving the other. Without the control logic, the buffer would latch itself in a logic low. When the B Side of the bus becomes active, the control logic disables the AB amplifier and activates the BA amplifier.

Note: *Pull-up resistors are required to provide logic high on all data and clock lines on both sides of the repeater, and on enable lines. Resistance values depend on the device and the speed of the system. Decoupling capacitors are also required and should be located as close to Vcc pins as possible.*

I²C Buffer Advantages

Bidirectional buffering is possible on both the serial data and serial clock lines. A multi-master bus design can position master devices on either side of the buffer. This allows any master on any bus segment to communicate with any slave on the bus. Negotiation for control of the bus is accomplished through software.

I²C buffers are available that support either unidirectional and bidirectional clock lines. Buffers can segment the bus and allow organization according to device specification or to isolate specific components. Each bus segment can be turned off and on as needed with the buffer enable. Buffers are also compatible with other I²C specifications such as the Power Management Bus (PMBUS) or System Management Bus (SMBUS).

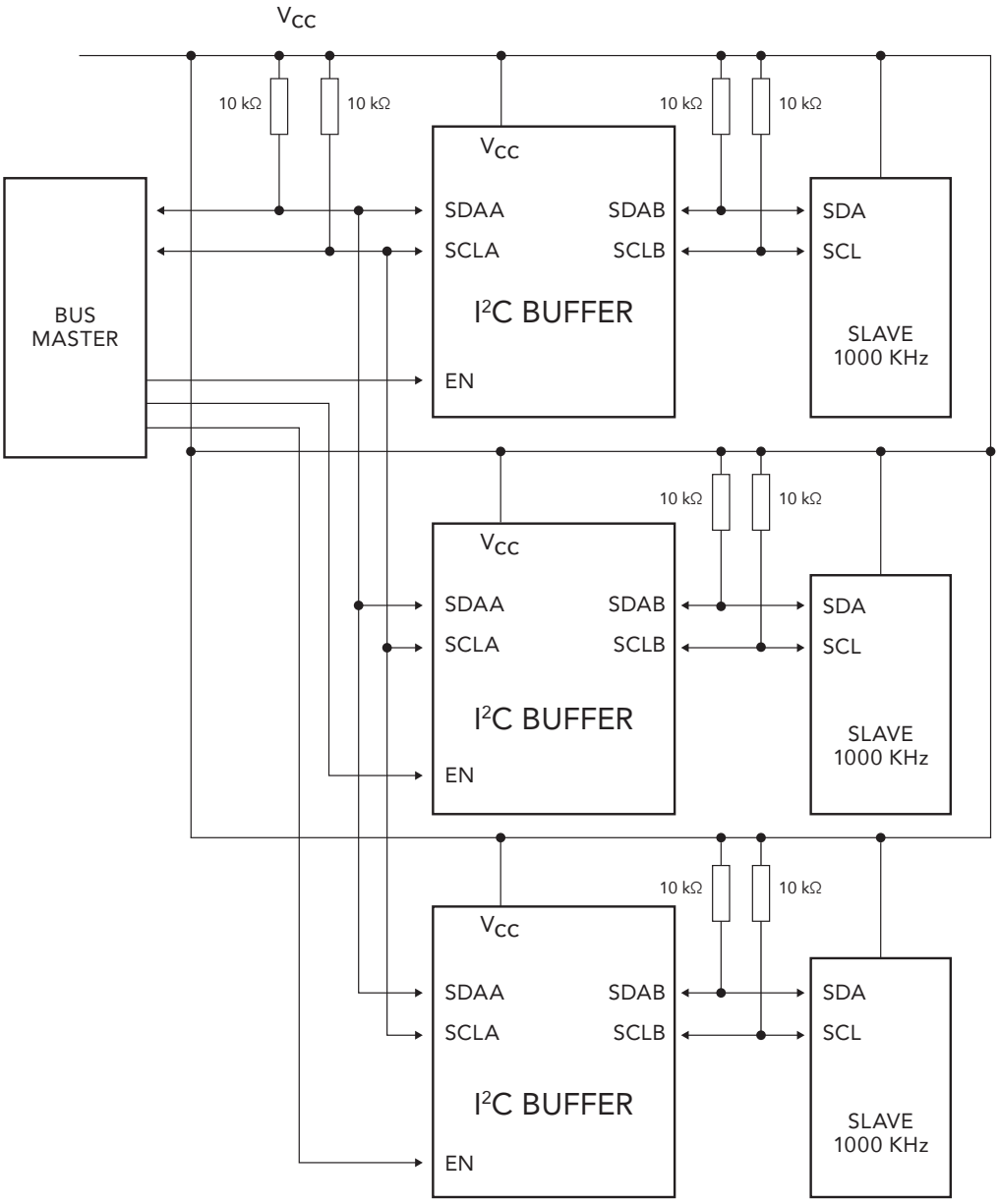
Implementation Configurations

The no-offset buffer repeats signals from one side of the buffer to the other. The A side and B side are isolated from each other and any offset in the serial data or serial clock signals is negated. Multiple buffers on the bus are allowed in either a star (parallel) or serial configuration, or to drive a cable between cards.

Star (Parallel)

The star configuration (Figure 2.) connects the A Side of each buffer together. The B side of each buffer device connects to the slaves located on that segment of the bus. The Master can turn each buffer on and off as needed with the enable pin, thus activating or deactivating any segment of the bus.

Figure 2. Typical Star Configuration



Level Translating Buffers

As the energy efficiency of some components increases, other components may not have the same capability. For example, a smart-phone application may wish to use an updated CPU that operates at a lower voltage than the components already on the bus. A level-translating buffer makes it possible for a CPU that operates at Vcc = 1.8V to work with other components at higher voltages such as Vcc = 3.3V or Vcc = 5.0V. Many new I²C Bus Buffers have two power supplies, and these are ideal for converting signals between two different voltage planes.

Minimize Capacitance

As the number of devices on the bus increase, the capacitance load also increases. I²C provides for a maximum capacitance loading of 400pF in standard mode, fast mode, and high speed mode up to 1.0MHz, but drops the allowed load to 100pF at speeds from 1.7MHz to 3.4MHz. Buffers segment the bus and isolate the capacitance of each segment from other segments. A small propagation delay (typically 150ns) from one segment to another is the only performance issue.

Hot Swap Cards

Many systems now require 24/7 up time. Such systems must not go offline when a single component fails or requires an update. System components are removed and added as necessary without taking down the entire system. Numerous applications include telecommunications, server farms, and drive bays. When a malfunctioning card or component is detected, the Master can disable the device until it is removed and replaced. Another possibility is a design that integrates redundancy. If a Master malfunctions, another Master can disable it and take over control of the bus. Buffers prevent bus corruption when a malfunctioning card is isolated and then removed. The replacement card waits for the bus and is activated only when it is ready to operate.

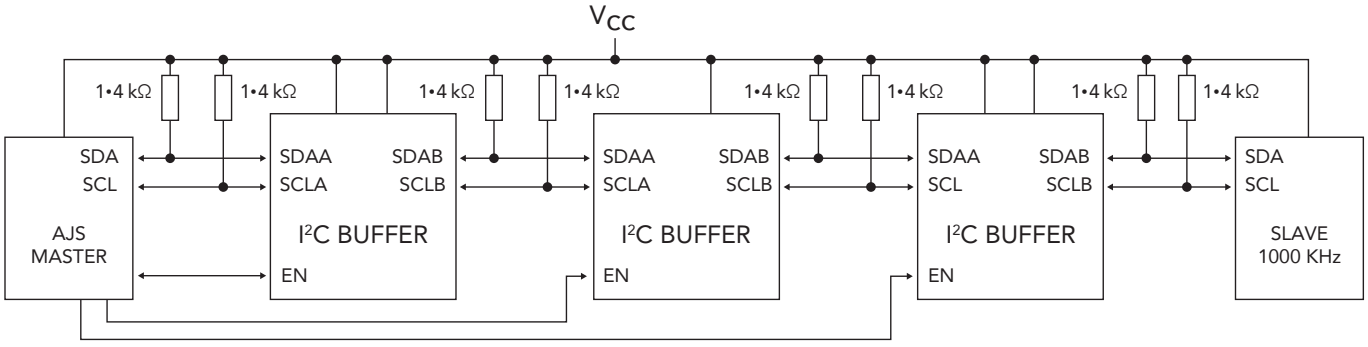
Group Components by Operational Requirements

Different components have different parameters for operation. Some may operate at one supply voltage and speed, while others are limited to slower speed at a higher voltage. The use of buffers allows a segmented bus design that segregates components by their operating specifications. Segmented design using I²C buffers allows integration of components regardless of their operating speed or voltage. Devices with slower operating speeds on their own bus segment are disabled to allow faster devices to operate. When the slower devices are needed, the Master can enable the buffer and use a slower clock to allow the slower segment to communicate with the faster segments. Another possibility allows a slower master to take control of the bus, generate its own clock, and thereby communicate with components that would normally operate at a higher speed.

Serial Configuration

The typical I²C Bus has a maximum length of a few meters. Buffers arranged in a serial configuration extend the bus length beyond the usual limitation. Each buffer IC extends the length by the maximum standalone bus length. Multiple buffers arranged this way can extend maximum length. Each buffer repeats the data and clock, negating any offset voltage.

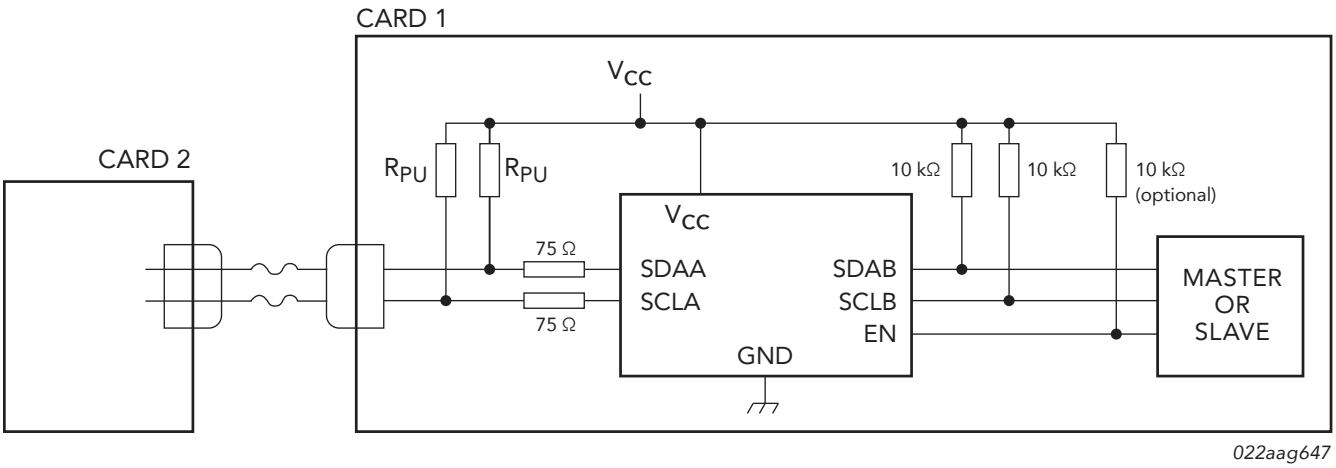
Figure 3. Typical Serial Configuration



Cable Driver

A common application places bus components on a separate card connected by a short cable. I²C buffers drive the cables to connect the bus components. Implementation scenarios include space requirements that necessitate two or more cards for ease of replacement.

Figure 4. Typical Card on a I²C Bus



Summary

The popular I²C bus continues to find its way into numerous applications from personal computers to consumer mobile devices, and from server farms to telecommunication equipment. The versatility of this I²C bus has enabled it to thrive since the early 1980s through its own evolution into a faster bus specification and through the addition of other, more specific applications such as system and power management.

As the number of I²C devices grows, the I²C buffer changes a potentially weak link into an advantage by allowing old and new components to effectively communicate with each other. The limitations of legacy components, increasingly high device counts, and limitations posed by speed considerations all make the I²C buffer a necessary and important part of any design specification.

Sources

AN10216-01 I2C Manual

PCA9617A Product Data Sheet