CP2120 PORTING GUIDE

Relevant Devices

This application note applies to the CP2120.

1. Introduction

The CP2120 SPI to SMBus bridge device can be used as a replacement for the Philips SC18IS60x device by making only a few minor hardware and firmware design modifications. All firmware and hardware differences between the SC18IS60x and CP2120 are outlined in Table 1.

Table 1. Improvement and Modification Summary

SCI18IS60x	CP2120	Section
Up to 6 general purpose I/O pins	8 general purpose I/O pins	2.1.1
96-byte buffers	256-byte buffers	2.1.2
	Ability to write a data buffer to up to 255 slave address	2.1.3
	Edge Triggered Interrupt Source	2.1.4
	Ability to read received byte buffer size	2.1.5
	SCL Low for SMBus compatibility	2.1.6
	Free Bus Detect for SMBus compatibility	2.1.6
No transition byte needed	Transition byte between SPI command write and read phases	2.1.1
Up to 3 MHz SPI frequency	Up to 1 MHz SPI frequency	2.1.2
	SMBus clock configuration	2.1.3
4.4 mm package size	4 mm package size	3

2. Firmware

The CP2120 responds to all SC18IS60x commands, including all Internal Register reads, writes, and all Internal Register reads and writes and all SMBus related commands. The CP2120 offers additional Internal Registers and commands that enhance performance of designs currently using the SC18IS60x device. However, designers must make certain modifications, which are discussed in the following sections, to their existing SPI Master interface in order to communicate with the CP2120.

2.1. Added Features

The CP2120 incorporates a number of features not found in the SC18IS60x. SPI Master firmware can take advantage of these added features to create a more efficient and reliable system.

2.1.1. 256-Byte Data Buffers

The CP2120 uses 256-byte data buffers to send and receive bytes with SMBus slaves. Calls to the Write Number of Bytes command can transmit up to 255 bytes in a single transaction, while calls to Read Number of Bytes can store up to 255 bytes of data into a buffer that can then be read by a Read Buffer command call.

The Write After Write command, which transmits to one slave address and then immediately transmits to a second slave address, can transmit any amount of data as long as the sum of the two buffers' sizes does not exceed 255 bytes. The Read After Write command can write 255 bytes and then immediately receive 255 bytes, which can then be retrieved by the SPI Master during a Read Buffer call.

The Write To Multiple Slaves command can write to any number of slaves any data buffer size, as long as the number of slaves listed in the command plus the number of bytes stored in the data buffer does not exceed 255.

2.1.2. 8 General Purpose I/O Pins

The CP2120 offers 8 general purpose input/output port pins that can be configured to either output or input using the IOCONFIG and IOCONFIG2 Internal Registers. Writes to the IOSTATE register update logic levels of port pins configured as outputs, while reading the IOSTATE register returns instantaneous port pin states.

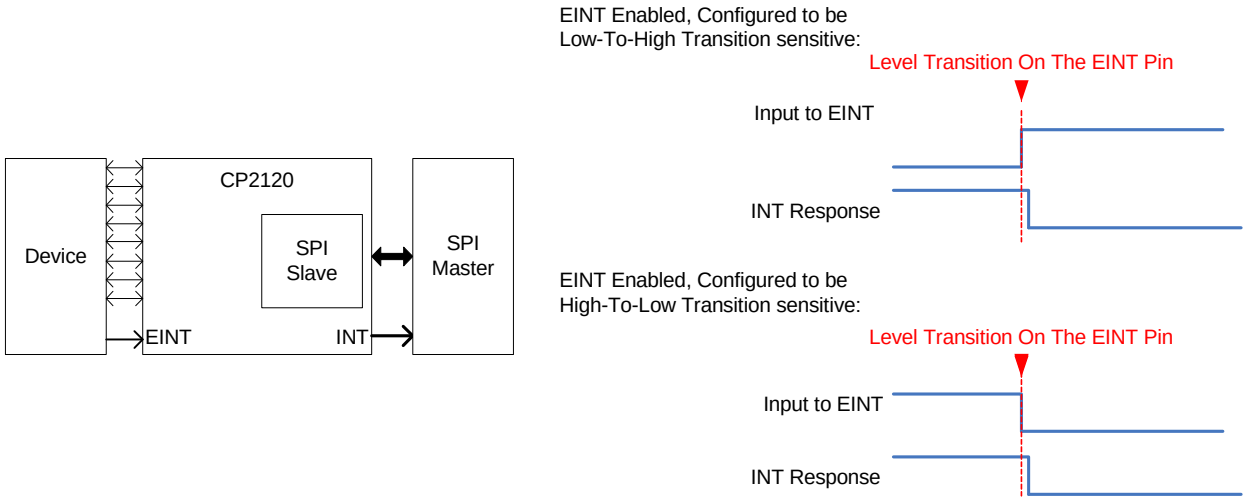


Figure 1. EINT Functionality

2.1.3. Write to Multiple Slaves Command

A SPI Master can transmit an identical data buffer to multiple slave devices using the Write to Multiple Slaves Command. Once the command, containing both a data buffer and a buffer of SMBus slave addresses, has been completely transmitted to the CP2120, the device attempts to transmit to each slave in the slave address buffer. If time-out timers configured through I2CTO and I2CTO2 are enabled, those timers will be initialized, monitored, and reset during each individual SMBus transfer. When a SMBus device fails to respond, a configured time-out occurs, or a transaction completes successfully, the CP2120 attempts to communicate with the next slave listed in the slave address buffer.

2.1.4. Edge-Triggered Interrupt Source

Figure 1 shows the operation of the EINT Edge-triggered interrupt pin. The EINT digital input pin can be enabled and configured as an edge-triggered interrupt source by writing to the EDGEINT internal register. The pin can be configured to lower the INT pin when either a low-to-high or a high-to-low logic transition occurs on the EINT pin. Reading the EDGEINT internal register after an edge triggered interrupt will raise the INT pin back to logic high.

2.1.5. Received Bytes Register

Bytes received through SMBus calls such as Read Number of Bytes are stored in a data buffer. At any time, the SPI Master can read the size of this data buffer by reading the RXBUFF register. Reading RXBUFF before calling Read Buffer can help to ensure that the correct number of bytes gets read from the buffer, without missing any bytes or reading too many and retrieving invalid data.

2.1.6. Added SMBus Time-Out Events

The CP2120 offers two additional timers, SCL Low time-out and Free Bus Detect, that, when enabled, allow the CP2120 to be fully compatible with SMBus specifications. The SCL Low time-out timer monitors the logic level of SCL. If SCL remains low for longer than 25 milliseconds during an SMBus transaction, the CP2120 terminates the transaction and indicates the termination source by setting the I2CSTAT status register to “SCL Low Time Out”. When the Free Bus Detect timer is enabled, the CP2120 considers SMBus bus lines free if SCL has remained high for at least 50 µs.

2.2. Modifications

Like the SC18IS60x device, the CP2120 receives commands across a SPI bus and then acts as an SMBus bus master to transmit and receive data with SMBus slaves. The SPI bus is controlled by a SPI master firmware system, and this system must be modified to ensure reliable communication.

2.2.1. Commands Requiring the CP2120 to Transmit Data Across SPI

The Read Internal Register, Read Buffer, and Revision Number commands cause the CP2120 to transmit data back to the SPI Master. During these commands, the SPI Master transmits command bytes through the MOSI SPI pin and then receives bytes through the MISO SPI pin. For these commands, the SPI Master must transmit one extra data byte after valid command bytes have been transmitted before valid bytes will be output from the CP2120.

Designers can make this modification to their SPI Master firmware systems in a number of ways. For example, if the SPI Master transmits commands by pulling bytes from an outgoing SPI data byte buffer, the design can push one extra byte onto the buffer after the system has pushed all valid bytes to transmit. When the command is transmitted to the CP2120, the extra byte will be transmitted as well.

2.2.2. SPI Clock Timing

The CP2120’s SPI interface can operate at SCLK frequencies up to 1 MHz. If the current, SC18IS60x-based design communicates at a SPI clock of frequencies higher than 1 MHz, the CP2120 will not be able to respond reliably with valid data.

2.2.3. I2CCLOCK Internal Register Values

While the CP2120 can reproduce the same range of SMBus clock frequencies as the SC18IS60x, the CP2120 uses a different equation than the SC18IS60x to map register values to SMBus SCL frequencies. For this reason, code configuring the SMBus clock frequency will need to be changed. The CP2120 uses the following equation to determine SMBus clock frequency:

$$\text{SMBus Clock Frequency (kHz)} = \frac{200}{\text{I2CCLOCK}}$$

3. Hardware

When porting a SC18IS60x design into a CP2120 design, the schematic and layout must be altered to take into account the CP2120’s pin-out and package size.

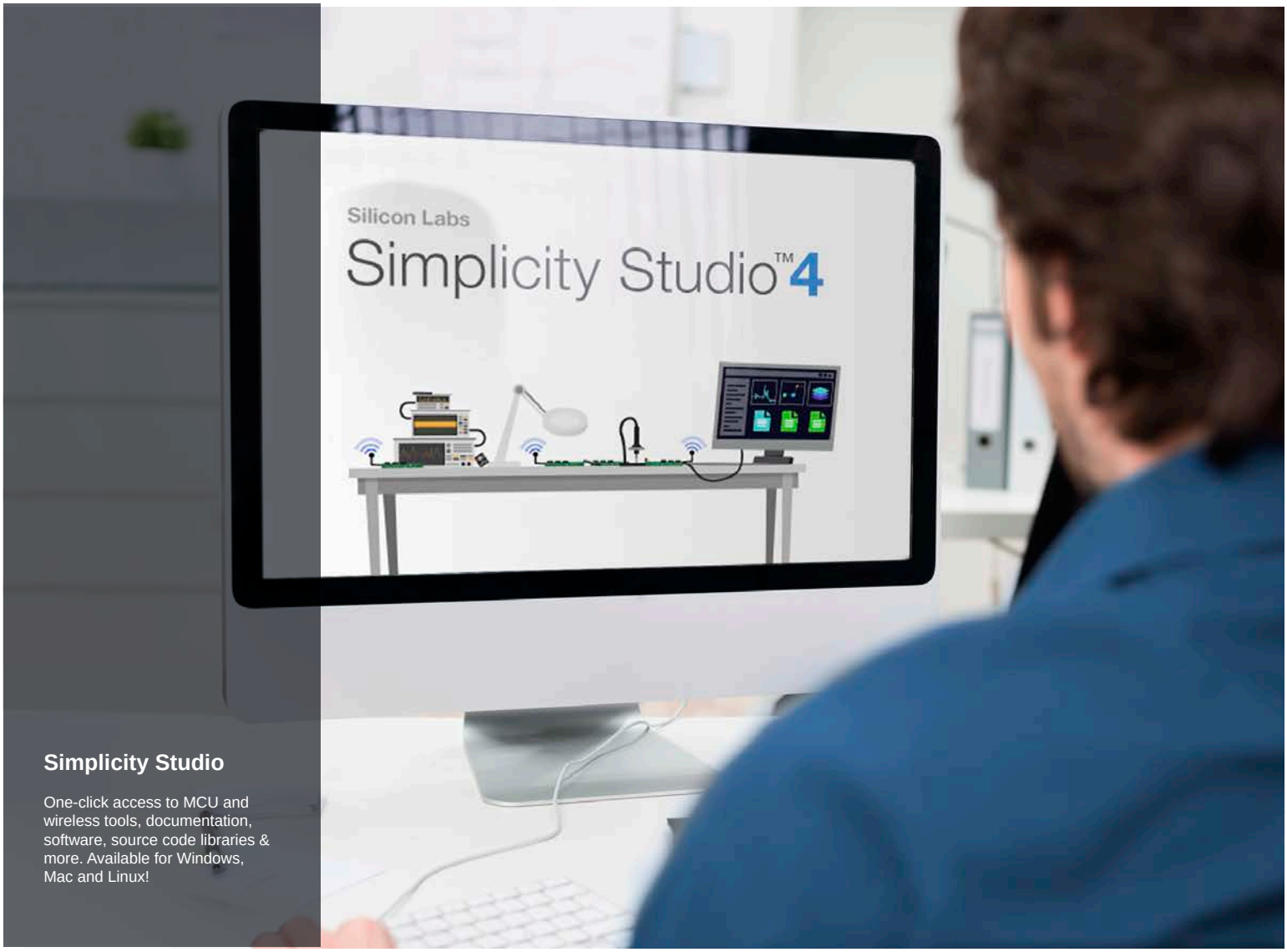
3.1. Schematic Modifications

While CP2120 and SC18IS60x pin names are identical, the CP2120 has more pins to accommodate the additional features. Updates to the schematic should include adding traces for the extra GPIO pins and the edge-triggered interrupt.

3.2. Layout Modifications

The CP2120’s pin-out is drastically different from the pin-out of the SC18IS60x, and updates to the layout must take into account the fact that many signal traces will need to be routed to different pins. For instance, the SC18IS60x SCLK trace is routed to pin 11, but the CP2120 routes that signal to pin 1.

The footprints for the SC18IS60x and the CP2120 also differ. The SC18IS60x uses a TSSOP16 package, while the CP2120 uses a smaller QFN-20 package.



Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

IoT Portfolio
www.silabs.com/IoT

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support and Community
community.silabs.com

SILICON LABS

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

http://www.silabs.com