



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China





CY7C65215
CY7C65215A

USB-Serial Dual Channel (UART/I²C/SPI) Bridge with CapSense[®] and BCD

Features

- USB 2.0 certified, Full-Speed (12 Mbps)
 - Support for communication driver class (CDC), personal health care device class (PHDC), and vendor device class
 - Battery charger detection (BCD) compliant with USB Battery Charging Specification Rev 1.2 (Peripheral Detect only)
 - Integrated USB termination resistors
 - Two-channel configurable UART interfaces
 - CY7C65215 supports 2-pin, 4-pin and 6-pin UART interface whereas CY7C65215A supports 2-pin, 4-pin, 6-pin and 8-pin UART interface
 - Data rates up to 3 Mbps
 - 190 bytes each transmit and receive buffer per channel
 - Data format:
 - 7 or 8 data bits
 - 1 or 2 stop bits
 - No parity, even, odd, mark, or space parity
 - Supports parity, overrun, and framing errors
 - Supports flow control using CTS, RTS, DTR, DSR
 - Supports UART break signal
 - CY7C65215 supports dual channel RS232/RS422 interfaces whereas CY7C65215A supports RS232/RS422/RS485 interfaces
 - Two-channel configurable SPI interfaces
 - Data rate up to 3 MHz for SPI master and 1 MHz for SPI slave
 - Data width: 4 bits to 16 bits
 - 256 bytes for each transmit and receive buffer per channel
 - Supports Motorola, TI, and National SPI modes
 - Two-channel configurable I²C interfaces
 - Master/slave up to 400 kHz
 - Supports multi-master I²C
 - 256 bytes for each transmit and receive buffer per channel
 - CapSense[®]
 - SmartSense™ Auto-Tuning is supported through a Cypress-supplied configuration utility
 - Max CapSense buttons: 8
 - GPIOs linked to CapSense buttons
 - JTAG interface: JTAG master for code flashing at 400 kHz
 - General-purpose input/output (GPIO) pins: 17
 - Supports unique serial number feature for each device, which fixes the COM port number permanently when USB-Serial Bridge controller as CDC device plugs in
 - Configuration utility (Windows) to configure the following:
 - Vendor ID (VID), Product ID (PID), and Product and Manufacturer descriptors
 - UART/I²C/SPI/JTAG
 - CapSense
 - Charger detection
 - GPIO
- Driver support for VCOM and DLL
 - Windows 10: 32- and 64-bit versions
 - Windows 8.1: 32- and 64-bit versions
 - Windows 8: 32- and 64-bit versions
 - Windows 7: 32- and 64-bit versions
 - Windows Vista: 32- and 64-bit versions
 - Windows XP: 32- and 64-bit versions
 - Windows CE
 - Mac OS-X: 10.6, and later versions
 - Linux: Kernel version 2.6.35 onwards
 - Android: Gingerbread and later versions
 - Clocking: Integrated 48-MHz clock oscillator
 - Supports bus-/self-powered configurations
 - USB suspend mode for low power
 - Operating voltage: 1.71 to 5.5 V
 - Operating temperature:
 - Commercial: 0 °C to 70 °C
 - Industrial: -40 °C to 85 °C
 - ESD protection: 2.2 kV HBM
 - RoHS compliant package
 - 32-pin QFN (5 × 5 × 1 mm, 0.5 mm pitch)
 - Ordering part number
 - CY7C65215-32LTXI
 - CY7C65215A-24LTXI

Applications

- Medical/healthcare devices
- Point-of-Sale (POS) terminals
- Test and measurement system
- Gaming systems
- Set-top box PC-USB interface
- Industrial
- Networking
- Enabling USB connectivity in legacy peripherals

USB Compliant

The USB-Serial Dual-Channel Bridge with CapSense and BCD (CY7C65215/CY7C65215A) is fully compliant with the USB 2.0 specification and Battery Charging Specification v1.2, USB-IF Test-ID (TID) 40001521.



CY7C65215 and CY7C65215A Features Comparison

Table 1. CY7C65215 and CY7C65215A Features Comparison

| Features | CY7C65215 | CY7C65215A |
|--------------------|--|--|
| UART | Can be configured as Virtual COM port or USB vendor device | Can be configured as Virtual COM port or USB vendor device |
| I ² C | Can be configured as USB vendor device | Can be configured as Virtual COM port or USB vendor device |
| SPI | Can be configured as USB vendor device | Can be configured as Virtual COM port or USB vendor device |
| RS485 Support | No | Yes |
| 8-pin UART Support | No | Yes |
| JTAG Support | Yes | No |

More Information

Cypress provides a wealth of data at www.cypress.com to help you to select the right device for your design, and to help you to quickly and effectively integrate the device into your design. For a comprehensive list of resources, see the document [USB-Serial Bridge Controller Product Overview](#).

- Overview: [USB Portfolio](#), [USB Roadmap](#)
- USB 2.0 Product Selectors: [USB-Serial Bridge Controller](#), [USB to UART Controller \(Gen I\)](#)
- Knowledge Base Articles: Cypress offers a large number of USB knowledge base articles covering a broad range of topics, from basic to advanced level. Recommended knowledge base articles for getting started with USB-Serial Bridge Controller are:
 - [KBA85909](#) – Key Features of the Cypress® USB-Serial Bridge Controller
 - [KBA85920](#) – USB-UART and USB-Serial
 - [KBA85921](#) – Replacing FT232R with CY7C65213 USB-UART LP Bridge Controller
 - [KBA85913](#) – Voltage supply range for USB-Serial
 - [KBA89355](#) – USB Serial Cypress Default VID and PID
 - [KBA92641](#) – USB-Serial Bridge Controller Managing I/Os using API
 - [KBA92442](#) – Non-Standard Baud Rates in USB-Serial Bridge Controllers
 - [KBA91366](#) – Binding a USB-Serial Device to a Microsoft® CDC Driver
 - [KBA92551](#) – Testing a USB-Serial Bridge Controller Configured as USB-UART with Linux®
 - [KBA91299](#) – Interfacing an External I2C Device with the CYUSBS234/236 DVK

For complete list of knowledge base articles, click [here](#).

- Code Examples: [USB Full-Speed](#)
- Development Kits:
 - [CYUSBS232](#), Cypress USB-UART LP Reference Design Kit
 - [CYUSBS234](#), Cypress USB-Serial (Single Channel) Development Kit
 - [CYUSBS236](#), Cypress USB-Serial (Dual Channel) Development Kit

■ Models: IBIS

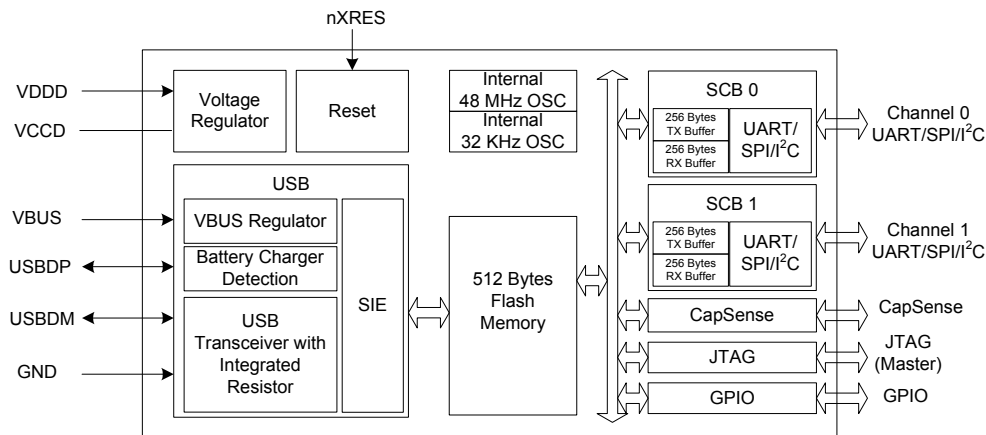
Cypress USB-Serial (Dual Channel) Development Kit

The [Cypress USB-Serial \(Dual Channel\) Development Kit](#) is a complete development resource. It provides a platform to develop and test custom projects. The development kit contains collateral materials for the firmware, hardware, and software aspects of a design.

Contents

| | | | |
|--|-----------|--|-----------|
| Block Diagram | 4 | USB Power Configurations | 18 |
| Functional Overview | 4 | USB Bus-Powered Configuration | 18 |
| USB and Charger Detect | 4 | Self-Powered Configuration | 19 |
| Serial Communication | 5 | USB Bus Powered with Variable I/O Voltage | 20 |
| CapSense | 6 | Application Examples | 21 |
| JTAG Interface | 6 | USB-to-Dual UART Bridge | |
| GPIO Interface | 6 | with Battery-Charge Detection | 21 |
| Memory | 6 | USB to RS485 Application | 23 |
| System Resources | 6 | CapSense | 24 |
| Suspend and Resume | 7 | USB to Dual Channel (I2C/SPI) Bridge | 25 |
| WAKEUP | 7 | Ordering Information | 30 |
| Software | 7 | Ordering Code Definitions | 30 |
| Internal Flash Configuration | 8 | Package Information | 31 |
| Electrical Specifications | 9 | Acronyms | 32 |
| Absolute Maximum Ratings | 9 | Document Conventions | 32 |
| Operating Conditions | 9 | Units of Measure | 32 |
| Device Level Specifications | 9 | Document History Page | 33 |
| GPIO | 10 | Sales, Solutions, and Legal Information | 35 |
| nXRES | 11 | Worldwide Sales and Design Support | 35 |
| SPI Specifications | 12 | Products | 35 |
| I2C Specifications | 14 | PSoC® Solutions | 35 |
| JTAG Specifications | 14 | Cypress Developer Community | 35 |
| CapSense Specifications | 14 | Technical Support | 35 |
| Flash Memory Specifications | 14 | | |
| Pin Description | 15 | | |

Block Diagram



Functional Overview

The CY7C65215/CY7C65215A is a Full-Speed USB controller that enables seamless PC connectivity for peripherals with dual-channel serial interfaces such as UART, SPI, and I²C. CY7C65215/CY7C65215A also integrates CapSense and BCD, which is compliant with the USB Battery Charging Specification Rev. 1.2. It integrates a voltage regulator, oscillator, and flash memory for storing configuration parameters, offering a cost-effective solution. CY7C65215/CY7C65215A supports bus-powered and self-powered modes, and enables efficient system power management with suspend and remote wake-up signals. It is available in a 32-pin QFN package.

USB and Charger Detect

USB

CY7C65215/CY7C65215A has a built-in USB 2.0 Full-Speed transceiver. The transceiver incorporates the internal USB series

termination resistors on the USB data lines and a 1.5-k Ω pull-up resistor on USBDP.

Charger Detection

CY7C65215/CY7C65215A supports BCD for Peripheral Detect only and complies with the USB Battery Charging Specification Rev. 1.2. It supports the following charging ports:

- Standard Downstream Port (SDP): allows the system to draw up to 500 mA current from the host
- Charging Downstream Port (CDP): allows the system to draw up to 1.5 A current from the host
- Dedicated Charging Port (DCP): allows the system to draw up to 1.5 A of current from the wall charger

Serial Communication

CY7C65215/CY7C65215A has two serial communication blocks (SCBs). Each SCB can implement UART, SPI, or an I²C interface. A 256-byte buffer is available in both the TX and RX lines.

Table 2 shows maximum speed supported on both SCBs when they are configured as UART/I²C/SPI.

Table 2. Maximum Speed supported on both SCBs

| No. | Configuration | SCB0 Maximum Speed | SCB1 Maximum Speed |
|-----|------------------------------------|--------------------------|--------------------------|
| 1 | SCB0 = UART, SCB1 = Disabled | 3M (Either TX/RX) | NA |
| 2 | SCB0 = I2C Master, SCB1 = Disabled | 400 kHz (Both TX and RX) | NA |
| 3 | SCB0 = I2C Slave, SCB1 = Disabled | 400 kHz (Both TX and RX) | NA |
| 4 | SCB0 = SPI Master, SCB1 = Disabled | 3M (Both TX and RX) | NA |
| 5 | SCB0 = SPI Slave, SCB1 = Disabled | 1M (Both TX and RX) | NA |
| 6 | SCB0 = UART, SCB1 = UART | 1M (Either TX/RX) | 1M (Either TX/RX) |
| 7 | SCB0 = UART, SCB1 = I2C Master | 1M (Either TX/RX) | 400 kHz (Both TX and RX) |
| | SCB0 = I2C Master, SCB1 = UART | | |
| 8 | SCB0 = UART, SCB1 = I2C Slave | 1M (Either TX/RX) | 400 kHz (Both TX and RX) |
| | SCB0 = I2C slave, SCB1 = UART | | |
| 9 | SCB0 = UART, SCB1 = SPI Master | 1M (Either TX/RX) | 1M (Both TX and RX) |
| | SCB1 = SPI Master, SCB0 = UART | | |
| 10 | SCB0 = UART, SCB1 = SPI Slave | 1M (Either TX/RX) | 1M (Both TX and RX) |
| | SCB0 = SPI Slave, SCB1 = UART | | |
| 11 | SCB0 = I2C, SCB1 = I2C | 400 kHz (Both TX and RX) | 400 kHz (Both TX and RX) |
| 12 | SCB0 = SPI, SCB1 = SPI | 1M (Both TX and RX) | 1M (Both TX and RX) |

UART Interface

The UART interface provides asynchronous serial communication with other UART devices operating at speeds of up to 3 Mbps. It supports 7 to 8 data bits, 1 to 2 stop bits, odd, even, mark, space, and no parity. The UART interface supports full duplex communication with a signaling format that is compatible with the standard UART protocol. In CY7C65215, UART pins may be interfaced to industry standard RS232/RS422 transceivers whereas in CY7C65215A these UART pins may be interfaced to RS232/RS422/RS485.

CY7C65215/CY7C65215A supports common UART functions such as parity error and frame error. In addition, CY7C65215/CY7C65215A supports baud rates ranging from 300 baud to 3 Mbaud. UART baud rates can be set using the configuration utility.

Notes:

Parity error gets detected when UART transmitter device is configured for odd parity and UART receiver device is configured for even parity.

Frame error gets detected when UART transmitter device is configured for 7 bits data width and 1 stop bit, whereas UART receiver device is configured for 8 bit data width and 2 stop bits.

UART Flow Control

The CY7C65215/CY7C65215A device supports UART hardware flow control using control signal pairs such as RTS# (Request to Send) / CTS# (Clear to Send) and DTR# (Data Terminal Ready) / DSR# (Data Set Ready). Data flow control is enabled by default. Flow control can be disabled using the configuration utility.

The following section describes the flow control signals:

■ CTS# (Input) / RTS# (Output)

CTS# can pause or resume data transmission over the UART interface. Data transmission can be paused by de-asserting the CTS signal and resumed with CTS# assertion. The pause and resume operation does not affect data integrity. With flow control enabled, receive buffer has a watermark level of 93%. After the data in the receive buffer reaches that level, the RTS# signal is de-asserted, instructing the transmitting device to stop data transmission. The start of data consumption by the application reduces device data backlog. When it reaches the 75% watermark level, the RTS# signal is asserted to resume data reception.

■ DSR# (Input) /DTR# (Output)

DSR#/DTR# signals are used to establish the communication link with the UART. These signals complement each other in their functionality, similar to CTS# and RTS#.

SPI Interface

The SPI interface supports SPI Master and SPI Slave. This interface supports the Motorola, TI, and National Microwire protocols. The maximum frequency of operation is 3 MHz in SPI master mode and 1 MHz in SPI slave mode. It can support transaction sizes ranging from 4 bits to 16 bits in length, SPI slave supports 4 bits to 8 bits and 12 bits to 16 bits data width at 1 MHz operation. Whereas, it supports 9 bits, 10 bits and 11 bits data width operation at 500 kHz operation (for more details, refer to [USB to Dual Channel \(I2C/SPI\) Bridge on page 25](#)).

I²C Interface

The I²C interface implements full multi-master/slave modes and supports up to 400 kHz. The configuration utility tool is used to set the I²C address in slave mode. This tool enables only even slave addresses. For further details on protocol, refer to the NXP I²C specification rev5.

Notes

- I²C ports are not tolerant of higher voltages and cannot be hot-swapped or powered up independently from the rest of the I²C system.
- The minimum fall time of the SCL is met (as per NXP I2C specification Rev. 5) when V_{DD} is between 1.71 V and 3.0 V. When V_{DD} is within the range of 3.0 V to 3.6 V, it is recommended to add a 50 pF capacitor on the SCL signal.

CapSense

CapSense functionality is supported on all the GPIO pins. Any GPIO pin can be configured as a sense pin (CS0–CS7) using the configuration utility. When implementing CapSense functionality, the GPIO_0 pin (configured as the modulator capacitor - Cmod) should be connected to ground through a 2.2-nF capacitor (see [Figure 12 on page 24](#)). CY7C65215/CY7C65215A supports SmartSense auto-tuning of CapSense parameters and does not require manual tuning. SmartSense auto-tuning compensates for printed circuit board (PCB) variations and device process variations.

Optionally, any GPIO pin can be configured as a Cshield and connected to the shield of the CapSense button as shown in [Figure 12 on page 24](#). The shield prevents false triggering of buttons due to water droplets and guarantees CapSense operation (sensors respond to finger touch). GPIOs can be linked to CapSense buttons to indicate the presence of a finger. CapSense functionality can be configured using configuration utility.

CY7C65215/CY7C65215A supports up to eight CapSense buttons. For more information on CapSense, refer to [Getting Started with CapSense](#).

JTAG Interface

CY7C65215/CY7C65215A supports a 5-pin JTAG in master mode for code flashing at 400 kHz.

Note: When JTAG is enabled, other interfaces in the CY7C65215/CY7C65215A device cannot be used.

GPIO Interface

CY7C65215/CY7C65215A has 17 GPIOs. A maximum of 17 GPIOs are available for configuration if one 2-pin (I²C/2-pin

UART) serial interface is implemented. The configuration utility allows configuration of the GPIO pins. The configurable options are as follows:

- TRISTATE: GPIO tristated
- DRIVE 1: Output static 1
- DRIVE 0: Output static 0
- POWER#: Power control for bus power designs
- TXLED#: Drives LED during USB transmit
- RXLED#: Drives LED during USB receive
- TX or RX LED#: Drives LED during USB transmit or receive
GPIO can be configured to drive LED at 8-mA drive strength.
- BCD0/BCD1: Two-pin output to indicate the type of USB charger
- BUSDETECT: Connects VBUS pin for USB host detection
- CS0–CS7: CapSense button input (Sense pin)
- CSout0–CSout3: Indicates which CapSense button is pressed
- Cmod: External modulator capacitor that connects a 2.2-nF capacitor (±10%) to ground (GPIO_0 only)
- Cshield: Shield for waterproofing

Memory

CY7C65215/CY7C65215A has a 512-byte flash. The flash is used to store the USB parameters such as VID/PID, serial number, Product, and Manufacturer Descriptors, which can be programmed by the configuration utility.

System Resources

Power System

CY7C65215/CY7C65215A supports the USB Suspend mode to control power usage. CY7C65215/CY7C65215A operates in bus-powered or self-powered modes over a range of 3.15 to 5.5 V.

Clock System

CY7C65215/CY7C65215A has a fully integrated clock and does not require any external components. The clock system is responsible for providing clocks to all subsystems.

Internal 48-MHz Oscillator

The internal 48-MHz oscillator is the primary source of internal clocking in CY7C65215/CY7C65215A.

Internal 32-kHz Oscillator

The internal 32-kHz oscillator is primarily used to generate clocks for peripheral operation in the USB Suspend mode.

Reset

The reset block ensures reliable power-on reset and brings the device back to the default known state. The nXRES (active low) pin can be used by external devices to reset the CY7C65215/CY7C65215A.

Suspend and Resume

The CY7C65215/CY7C65215A device asserts the SUSPEND pin when the USB bus enters the suspend state. This helps in meeting the stringent suspend current requirement of the USB 2.0 specification, while using the device in bus-powered mode. The device will resume from the suspend state under any of the following conditions:

1. Any activity is detected on the USB bus
2. The WAKEUP pin is asserted to generate remote wakeup to the host

WAKEUP

The WAKEUP pin is used to generate a remote wakeup signal on the USB bus. The remote wakeup signal is sent only if the host enables this feature through the SET_FEATURE request. The device communicates support for the remote wakeup to the host through the configuration descriptor during the USB enumeration process. The CY7C65215/CY7C65215A device allows enabling/disabling and polarity of the remote wakeup feature through the configuration utility.

Software

Cypress delivers a complete set of software drivers and the configuration utility to enable product configuration during system development.

Drivers for Linux Operating Systems

Cypress provides a User Mode USB driver library (*libcyusb-serial.so*) that abstracts vendor commands for the UART interface and provides a simplified API interface to the user applications. This library makes use of the standard open source libUSB library to enable the USB communication. The Cypress serial library supports the USB plug-and-play feature using the Linux 'udev' mechanism.

CY7C65215/CY7C65215A supports the standard USB CDC UART class driver, which is bundled with the Linux kernel.

Android Support

The CY7C65215/CY7C65215A solution includes an Android Java class—*CyUsbSerial.java*—which exposes a set of interface functions to communicate with the device.

Drivers for Mac OSx

Cypress delivers a dynamically linked shared library (*CyUSBSerial.dylib*) based on libUSB, which enables communication to the CY7C65215/CY7C65215A device.

In addition, the CY7C65215 device also supports the native Mac OSx CDC UART driver, and CY7C65215A supports native Mac OSx CDC UART/SPI/I2C driver.

Drivers for Windows Operating Systems

For Windows operating systems (XP, Vista, Win7, Win8, and Win8.1), Cypress delivers a User Mode dynamically linked library—*CyUSBSerial DLL*—that abstracts vendor-specific interface of CY7C65215/CY7C65215A devices and provides convenient APIs to the user. It provides interface APIs for vendor-specific UART/SPI/I2C and class-specific APIs for PHDC.

USB-Serial Bridge Controller works with the Windows-standard USB CDC class driver, when either CY7C65215 is configured as CDC USB to UART device or when CY7C65215A is configured as CDC USB to UART/SPI/I2C device. A virtual COM port driver—*CyUSBSerial.sys*—is also delivered, which implements the USB CDC class driver. The Cypress Windows drivers are Windows hardware certification kit-compliant.

These drivers are bound to device through WU (Windows Update) services.

Cypress drivers also support Windows plug-and-play and power management and USB Remote Wake-up.

Windows-CE support

The CY7C65215/CY7C65215A solution includes a CDC UART driver library for Windows-CE platforms.

Device Configuration Utility (Windows Only)

A Windows-based configuration utility is available to configure various device initialization parameters. This graphical user application provides an interactive interface to define the various boot parameters stored in the device flash.

This utility allows the user to save a user-selected configuration to text or xml formats. It also allows users to load a selected configuration from text or xml formats. The configuration utility allows the following operations:

- View current device configuration
- Select and configure UART/I2C/SPI, CapSense, battery charging, and GPIOs
- Configure USB VID, PID, and string descriptors
- Save or Load configuration

You can download the free configuration utility and drivers from www.cypress.com.

Internal Flash Configuration

The internal flash memory can be used to store the configuration parameters shown in the following table. A free configuration utility is provided to configure the parameters listed in the table to meet application specific requirements over USB interface. The configuration utility can be downloaded from www.cypress.com/go/usbserial.

Table 3. Internal Flash Configuration for both CY7C65215 and CY7C65215A

| Parameter | Default Value | Description |
|------------------------|---------------------------|---|
| USB Configuration | | |
| USB Vendor ID (VID) | 0x04B4 | Default Cypress VID. Can be configured to customer VID |
| USB Product ID (PID) | 0x0005 | Default Cypress PID. Can be configured to customer PID |
| Manufacturer string | Cypress | Can be configured with any string up to 64 characters |
| Product string | USB-Serial (Dual Channel) | Can be configured with any string up to 64 characters |
| Serial string | | Can be configured with any string up to 64 characters |
| Power mode | Bus powered | Can be configured to bus-powered or self-powered mode |
| Max current draw | 100 mA | Can be configured to any value from 0 to 500 mA. Based on this, the configuration descriptor will be updated. |
| Remote wakeup | Enabled | Can be disabled. Remote wakeup is initiated by asserting WAKEUP pin |
| USB interface protocol | CDC | Can be configured to function in CDC, PHDC, or Cypress vendor class |
| BCD | Disabled | Charger detect is disabled by default. When BCD is enabled, three of the GPIOs must be configured for BCD |
| GPIO Configuration | | |
| GPIO_0 | TXLED# | GPIO can be configured as shown in Table 18 on page 17 . |
| GPIO_1 | RXLED# | |
| GPIO_2 | DSR#_0 | |
| GPIO_3 | RTS#_0 | |
| GPIO_4 | CTS#_0 | |
| GPIO_5 | TxD_0 | |
| GPIO_6 | POWER# | |
| GPIO_7 | TRISTATE | |
| GPIO_8 | RxD_0 | |
| GPIO_9 | DTR#_0 | |
| GPIO_10 | RxD_1 | |
| GPIO_11 | TxD_1 | |
| GPIO_12 | RTS#_1 | |
| GPIO_13 | CTS#1 | |
| GPIO_14 | DSR#_1 | |
| GPIO_15 | DTR#_1 | |
| GPIO_16 | TRISTATE | |
| GPIO_17 | TRISTATE | |
| GPIO_18 | TRISTATE | |

Electrical Specifications

Absolute Maximum Ratings

Exceeding maximum ratings ^[1] may shorten the useful life of the device.

| | |
|--|--------------------------|
| Storage temperature | -55 °C to +100 °C |
| Ambient temperature with power supplied (Industrial) | -40 °C to +85 °C |
| Supply voltage to ground potential | |
| V _{DDD} | 6.0 V |
| V _{BUS} | 6.0 V |
| V _{CCD} | 1.95 V |
| V _{GPIO} | V _{DDD} + 0.5 V |

Static discharge voltage ESD protection levels:

■ 2.2-kV HBM per JESD22-A114

Latch-up current 140 mA

Current per GPIO 25 mA

Operating Conditions

T_A (ambient temperature under bias)

Industrial -40 °C to +85 °C

V_{BUS} supply voltage 3.15 V to 5.25 V

V_{DDD} supply voltage 1.71 V to 5.50 V

V_{CCD} supply voltage 1.71 V to 1.89 V

Device Level Specifications

All specifications are valid for -40 °C ≤ T_A ≤ 85 °C, T_J ≤ 100 °C, and 1.71 V to 5.50 V, except where noted.

Table 4. DC Specifications

| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|------------------|-----------------------------------|------|------|------|-------|---|
| V _{BUS} | V _{BUS} supply voltage | 3.15 | 3.30 | 3.45 | V | Set and configure correct voltage range using the configuration utility for V _{BUS} . |
| | | 4.35 | 5.00 | 5.25 | V | |
| V _{DDD} | V _{DDD} supply voltage | 1.71 | 1.80 | 1.89 | V | Used to set I/O and core voltage. Set and configure correct voltage range using the configuration utility for V _{DDD} . |
| | | 2.0 | 3.3 | 5.5 | V | |
| V _{CCD} | Output voltage (for core logic) | - | 1.80 | - | V | Do not use this supply to drive external device. <ul style="list-style-type: none"> 1.71 V ≤ V_{DDD} ≤ 1.89 V: Short the V_{CCD} pin with the V_{DDD} pin V_{DDD} > 2 V – connect a 1-μF capacitor (Cefc) between the V_{CCD} pin and ground |
| Cefc | External regulator voltage bypass | 1.00 | 1.30 | 1.60 | μF | X5R ceramic or better |
| I _{DD1} | Operating supply current | - | 13 | 18 | mA | USB 2.0 FS, UART at 1 Mbps single channel, no GPIO switching at V _{BUS} = 5 V, V _{DDD} = 5 V |
| I _{DD2} | USB Suspend supply current | - | 5 | - | μA | Does not include current through a pull-up resistor on USB DP. In USB suspend mode, the D+ voltage can go up to a maximum of 3.8 V. |

Table 5. AC Specifications

| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|-----------|------------------------------|-----|-----|-----|-------|--------------------|
| Zout | USB driver output impedance | 28 | - | 44 | Ω | |
| Twakeup | Wakeup from USB Suspend mode | - | 25 | - | μs | |

Note

- Usage above the absolute maximum conditions may cause permanent damage to the device. Exposure to Absolute Maximum conditions for extended periods of time may affect device reliability. When used below Absolute Maximum conditions but above normal operating conditions the device may not operate to specification.

GPIO
Table 6. GPIO DC Specification

| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|----------------|--|----------------------|-----|---------------------|------------|--|
| $V_{IH}^{[2]}$ | Input voltage high threshold | $0.7 \times V_{DD}$ | – | – | V | CMOS Input |
| V_{IL} | Input voltage low threshold | – | – | $0.3 \times V_{DD}$ | V | CMOS Input |
| $V_{IH}^{[2]}$ | LVTTL input, $V_{DD} < 2.7$ V | $0.7 \times V_{DD}$ | – | – | V | |
| V_{IL} | LVTTL input, $V_{DD} < 2.7$ V | – | – | $0.3 \times V_{DD}$ | V | |
| $V_{IH}^{[2]}$ | LVTTL input, $V_{DD} \geq 2.7$ V | 2 | – | – | V | |
| V_{IL} | LVTTL input, $V_{DD} \geq 2.7$ V | – | – | 0.8 | V | |
| V_{OH} | CMOS output voltage high level | $V_{DD} - 0.4$ | – | – | V | $I_{OH} = 4$ mA, $V_{DD} = 5$ V +/- 10% |
| V_{OH} | CMOS output voltage high level | $V_{DD} - 0.6$ | – | – | V | $I_{OH} = 4$ mA, $V_{DD} = 3.3$ V +/- 10% |
| V_{OH} | CMOS output voltage high level | $V_{DD} - 0.5$ | – | – | V | $I_{OH} = 1$ mA, $V_{DD} = 1.8$ V +/- 5% |
| V_{OL} | CMOS output voltage low level | – | – | 0.4 | V | $I_{OL} = 8$ mA, $V_{DD} = 5$ V +/- 10% |
| V_{OL} | CMOS output voltage low level | – | – | 0.6 | V | $I_{OL} = 8$ mA, $V_{DD} = 3.3$ V +/- 10% |
| V_{OL} | CMOS output voltage low level | – | – | 0.6 | V | $I_{OL} = 4$ mA, $V_{DD} = 1.8$ V +/- 5% |
| Rpullup | Pull-up resistor | 3.5 | 5.6 | 8.5 | k Ω | |
| Rpulldown | Pull-down resistor | 3.5 | 5.6 | 8.5 | k Ω | |
| I_{IL} | Input leakage current (absolute value) | – | – | 2 | nA | 25 °C, $V_{DD} = 3.0$ V |
| C_{IN} | Input capacitance | – | – | 7 | pF | |
| Vhysttl | Input hysteresis LVTTL; $V_{DD} > 2.7$ V | 25 | 40 | – | mV | |
| Vhyscmos | Input hysteresis CMOS | $0.05 \times V_{DD}$ | – | – | mV | |

Table 7. GPIO AC Specification

| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|-----------------|------------------------|-----|-----|-----|-------|--|
| $T_{RiseFast1}$ | Rise Time in Fast mode | 2 | – | 12 | ns | $V_{DD} = 3.3$ V/ 5.5 V, Load = 25 pF |
| $T_{FallFast1}$ | Fall Time in Fast mode | 2 | – | 12 | ns | $V_{DD} = 3.3$ V/ 5.5 V, Load = 25 pF |
| $T_{RiseSlow1}$ | Rise Time in Slow mode | 10 | – | 60 | ns | $V_{DD} = 3.3$ V/ 5.5 V, Load = 25 pF |
| $T_{FallSlow1}$ | Fall Time in Slow mode | 10 | – | 60 | ns | $V_{DD} = 3.3$ V/ 5.5 V, Load = 25 pF |
| $T_{RiseFast2}$ | Rise Time in Fast mode | 2 | – | 20 | ns | $V_{DD} = 1.8$ V, Load = 25 pF |
| $T_{FallFast2}$ | Fall Time in Fast mode | 20 | – | 100 | ns | $V_{DD} = 1.8$ V, Load = 25 pF |
| $T_{RiseSlow2}$ | Rise Time in Slow mode | 2 | – | 20 | ns | $V_{DD} = 1.8$ V, Load = 25 pF |
| $T_{FallSlow2}$ | Fall Time in Slow mode | 20 | – | 100 | ns | $V_{DD} = 1.8$ V, Load = 25 pF |

Note

 2. V_{IH} must not exceed $V_{DD} + 0.2$ V.

nXRES
Table 8. nXRES DC Specifications

| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|----------------------|------------------------------|------------------------|-----|------------------------|-------|--------------------|
| V _{IH} | Input voltage high threshold | 0.7 × V _{DDD} | – | – | V | |
| V _{IL} | Input voltage low threshold | – | – | 0.3 × V _{DDD} | V | |
| R _{pullup} | Pull-up resistor | 3.5 | 5.6 | 8.5 | kΩ | |
| C _{IN} | Input capacitance | – | 5 | – | pF | |
| V _{hysxres} | Input voltage hysteresis | – | 100 | – | mV | |

Table 9. nXRES AC Specifications

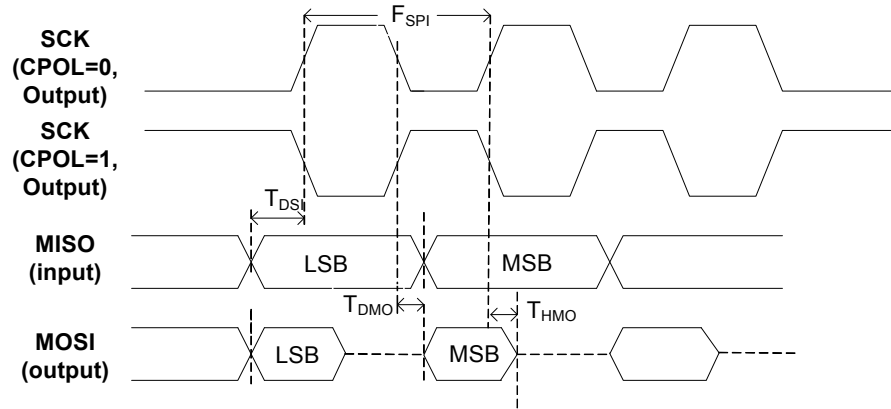
| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|-------------------------|-------------------|-----|-----|-----|-------|--------------------|
| T _{resetwidth} | Reset pulse width | 1 | – | – | μs | |

Table 10. UART AC Specifications

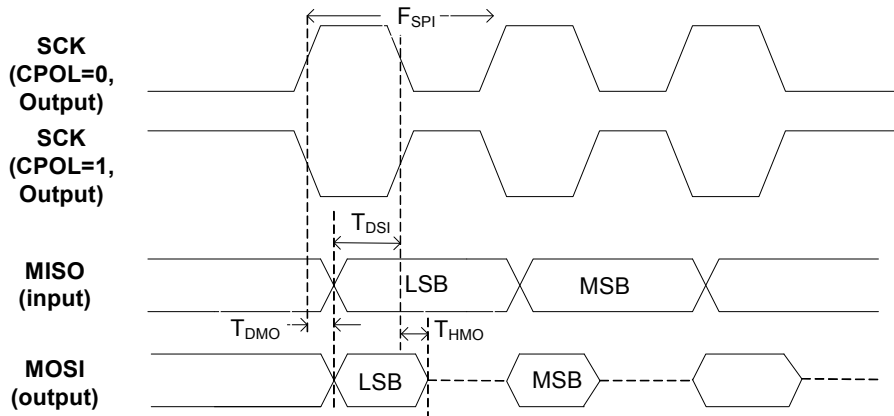
| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|-------------------|---------------|-----|-----|------|-------|---|
| F _{UART} | UART bit rate | 0.3 | – | 3000 | kbps | Single SCB: TX + RX Dual SCB: TX or RX |

SPI Specifications

Figure 1. SPI Master Timing

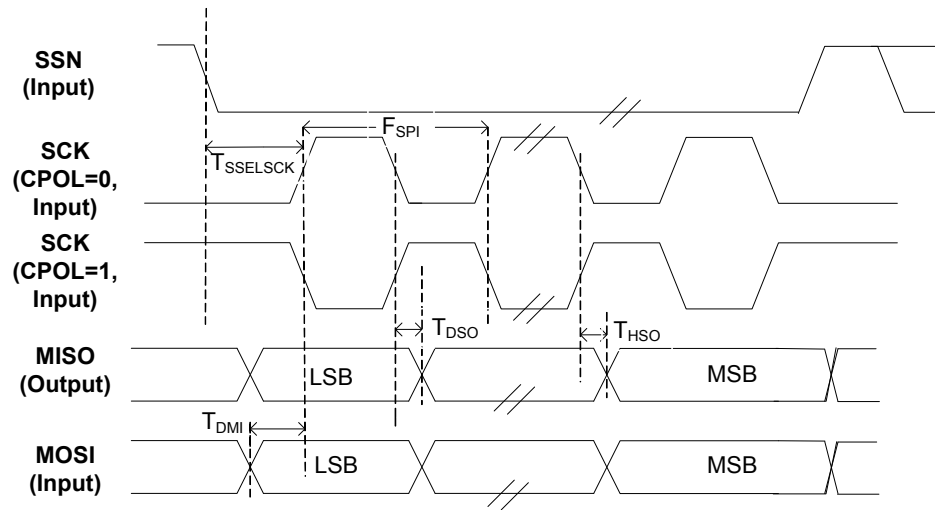


SPI Master Timing for CPHA = 0 (Refer to Table 17)

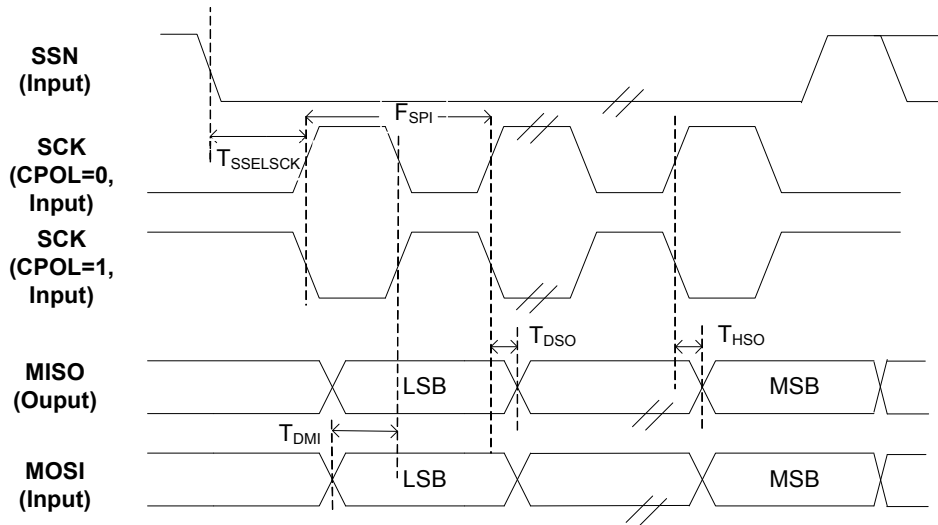


SPI Master Timing for CPHA = 1 (Refer to Table 17)

Figure 2. SPI Slave Timing



SPI Slave Timing for CPHA = 0 (Refer to Table 17)



SPI Slave Timing for CPHA = 1 (Refer to Table 17)

Table 11. SPI AC Specifications

| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|------------------------|--|-----|-----|-------|-------|---|
| F _{SPI} | SPI operating frequency (Master/Slave) | – | – | 3 | MHz | Single SCB: TX + RX Dual SCB: TX or RX |
| WL _{SPI} | SPI word length | 4 | – | 16 | bits | |
| SPI Master Mode | | | | | | |
| T _{DMO} | MOSI valid after SClock driving edge | – | – | 15 | ns | |
| T _{DSI} | MISO valid before SClock capturing edge | 20 | – | – | ns | |
| T _{HMO} | Previous MOSI data hold time with respect to capturing edge at slave | 0 | – | – | ns | |
| SPI Slave Mode | | | | | | |
| T _{DMI} | MOSI valid before Sclock Capturing edge | 40 | – | – | ns | |
| T _{DSO} | MISO valid after Sclock driving edge | – | – | 104.4 | ns | |
| T _{HSO} | Previous MISO data hold time | 0 | – | – | ns | |
| T _{SSELCK} | SSEL valid to first SCK valid edge | 100 | – | – | ns | |

I²C Specifications

Table 12. I²C AC Specifications

| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|------------------|----------------------------|-----|-----|-----|-------|--------------------|
| F _{I2C} | I ² C frequency | 1 | – | 400 | kHz | |

JTAG Specifications

Table 13. JTAG AC Specifications

| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|-------------------|-----------------------------------|-----|-----|-----|-------|--------------------|
| F _{JTAG} | JTAG operating frequency (master) | – | – | 400 | kHz | Code flashing |

CapSense Specifications

Table 14. CapSense AC Specifications

| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|------------------|------------------------------------|------|-----|------|-------|--|
| V _{CSD} | Voltage range of operation | 1.71 | – | 5.50 | V | |
| SNR | Ratio of counts of finger to noise | 5 | – | – | Ratio | Sensor capacitance range of 9 to 35 pF; finger capacitance ≥ 0.1 pF sensitivity |

Flash Memory Specifications

Table 15. Flash Memory Specifications

| Parameter | Description | Min | Typ | Max | Units | Details/Conditions |
|------------------|---|-------|-----|-----|--------|--------------------|
| F _{end} | Flash endurance | 100 K | – | – | cycles | |
| F _{ret} | Flash retention. T _A \leq 85 °C, 10 K program/erase cycles | 10 | – | – | years | |

Pin Description

| Pin ^[3] | Type | Name | | Default | Description |
|--------------------|----------|---------|---------|----------|--|
| 1 | Power | VDDD | | – | Supply to the device core and Interface, 1.71 to 5.5 V |
| 2 | SCB/GPIO | SCB0_0 | GPIO_8 | RxD_0 | GPIO/SCB0. See Table 16 and Table 18 on page 17 |
| 3 | SCB/GPIO | SCB0_5 | GPIO_9 | DTR#_0 | GPIO/SCB0. See Table 16 and Table 18 on page 17 |
| 4 | Power | VSSD | | – | Digital Ground |
| 5 | SCB/GPIO | SCB1_0 | GPIO_10 | RxD_1 | GPIO/SCB1. See Table 17 and Table 18 on page 17 |
| 6 | SCB/GPIO | SCB1_1 | GPIO_11 | TxD_1 | GPIO/SCB1. See Table 17 and Table 18 on page 17 |
| 7 | SCB/GPIO | SCB1_2 | GPIO_12 | RTS#_1 | GPIO/SCB1. See Table 17 and Table 18 on page 17 |
| 8 | SCB/GPIO | SCB1_3 | GPIO_13 | CTS#_1 | GPIO/SCB1. See Table 17 and Table 18 on page 17 |
| 9 | SCB/GPIO | SCB1_4 | GPIO_14 | DSR#_1 | GPIO/SCB1. See Table 17 and Table 18 on page 17 |
| 10 | SCB/GPIO | SCB1_5 | GPIO_15 | DTR#_1 | GPIO/SCB1. See Table 17 and Table 18 on page 17 |
| 11 | Output | SUSPEND | | – | Indicates device in suspend mode. Can be configured as active low/high using configuration utility |
| 12 | Input | WAKEUP | | – | Wakeup device from suspend mode. Can be configured as active low/high using configuration utility |
| 13 | GPIO | GPIO_16 | | TRISTATE | GPIO. See Table 18 on page 17 |
| 14 | USBIO | USBDP | | – | USB Data Signal Plus, integrates termination resistor and 1.5-kΩ pull up resistor |
| 15 | USBIO | USBDM | | – | USB Data Signal Minus, integrates termination resistor |
| 16 | Power | VCCD | | – | Regulated supply, connect to 1-μF cap or 1.8 V |
| 17 | Power | VSSD | | – | Digital Ground |
| 18 | nXRES | nXRES | | – | Chip reset, active low. Can be left unconnected or have a pull-up resistor connected if not used. |
| 19 | Power | VBUS | | – | VBUS Supply, 3.15 V to 5.25 V |
| 20 | Power | VSSD | | – | Digital Ground |
| 21 | GPIO | GPIO_17 | | TRISTATE | GPIO. See Table 18 on page 17 |
| 22 | GPIO | GPIO_18 | | TRISTATE | GPIO. See Table 18 on page 17 |
| 23 | Power | VDDD | | – | Supply to the device core and Interface, 1.71 to 5.5 V |
| 24 | Power | VSSA | | – | Analog Ground |
| 25 | GPIO | GPIO_0 | | TXLED# | GPIO. See Table 18 on page 17 |
| 26 | GPIO | GPIO_1 | | RXLED# | GPIO. See Table 18 on page 17 |
| 27 | SCB/GPIO | SCB0_1 | GPIO_2 | DSR#_0 | GPIO/SCB0. See Table 16 and Table 18 on page 17 |
| 28 | SCB/GPIO | SCB0_2 | GPIO_3 | RTS#_0 | GPIO/SCB0. See Table 16 and Table 18 on page 17 |
| 29 | SCB/GPIO | SCB0_3 | GPIO_4 | CTS#_0 | GPIO/SCB0. See Table 16 and Table 18 on page 17 |
| 30 | SCB/GPIO | SCB0_4 | GPIO_5 | TxD_0 | GPIO/SCB0. See Table 16 and Table 18 on page 17 |
| 31 | GPIO | GPIO_6 | | POWER# | GPIO. See Table 18 on page 17 |
| 32 | GPIO | GPIO_7 | | TRISTATE | GPIO. See Table 18 on page 17 |

Note

3. Any pin acting as an Input pin should not be left unconnected.

Figure 3. 32-Pin QFN Pinout

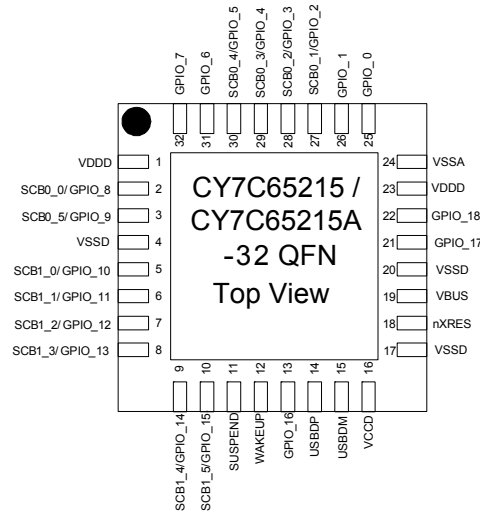


Table 16. Serial Communication Block (SCB0) Configuration

| Pin | Serial Port 0 | Mode 0* | Mode 1 | Mode 2 | Mode 3 | Mode 4 | Mode 5 | Mode 6 |
|-----|---------------|------------|------------|------------|------------|------------|-------------------------|------------------------|
| | | 6-pin UART | 4-pin UART | 2-pin UART | SPI Master | SPI Slave | I ² C Master | I ² C Slave |
| 2 | SCB0_0 | RxD_0 | RxD_0 | RxD_0 | GPIO_8 | GPIO_8 | GPIO_8 | GPIO_8 |
| 27 | SCB0_1 | DSR#_0 | GPIO_2 | GPIO_2 | SSEL_OUT_0 | SSEL_IN_0 | GPIO_2 | GPIO_2 |
| 28 | SCB0_2 | RTS#_0 | RTS#_0 | GPIO_3 | MISO_IN_0 | MISO_OUT_0 | SCL_OUT_0 | SCL_IN_0 |
| 29 | SCB0_3 | CTS#_0 | CTS#_0 | GPIO_4 | MOSI_OUT_0 | MOSI_IN_0 | SDA_0 | SDA_0 |
| 30 | SCB0_4 | TxD_0 | TxD_0 | TxD_0 | SCLK_OUT_0 | SCLK_IN_0 | GPIO_5 | GPIO_5 |
| 3 | SCB0_5 | DTR#_0 | GPIO_9 | GPIO_9 | GPIO_9 | GPIO_9 | GPIO_9 | GPIO_9 |

*Note: Device configured in Mode 0 as default. Other modes can be configured through Cypress-supplied configuration utility.

Table 17. Serial Communication Block (SCB1) Configuration

| Pin | Serial Port 1 | Mode 0* | Mode 1 | Mode 2 | Mode 3 | Mode 4 | Mode 5 | Mode 6 | Mode 7 |
|-----|---------------|------------|------------|------------|------------|------------|-------------------------|------------------------|-------------|
| | | 6-pin UART | 4-pin UART | 2-pin UART | SPI Master | SPI Slave | I ² C Master | I ² C Slave | JTAG Master |
| 5 | SCB1_0 | RxD_1 | RxD_1 | RxD_1 | MISO_IN_1 | MISO_OUT_1 | SCL_OUT_1 | SCL_IN_1 | TDO |
| 6 | SCB1_1 | TxD_1 | TxD_1 | TxD_1 | MOSI_OUT_1 | MOSI_IN_1 | SDA_1 | SDA_1 | TDI |
| 7 | SCB1_2 | RTS#_1 | RTS#_1 | GPIO_12 | SSEL_OUT_1 | SSEL_IN_1 | GPIO_12 | GPIO_12 | TMS |
| 8 | SCB1_3 | CTS#_1 | CTS#_1 | GPIO_13 | SCLK_OUT_1 | SCLK_IN_1 | GPIO_13 | GPIO_13 | TCK |
| 9 | SCB1_4 | DSR#_1 | GPIO_14 | GPIO_14 | GPIO_14 | GPIO_14 | GPIO_14 | GPIO_14 | TRST# |
| 10 | SCB1_5 | DTR#_1 | GPIO_15 | GPIO_15 | GPIO_15 | GPIO_15 | GPIO_15 | GPIO_15 | GPIO_15 |

*Note: Device configured in Mode 0 as default. Other modes can be configured via Cypress-supplied configuration utility.



Table 18. GPIO Configuration

| GPIO Configuration Option | Description |
|---|--|
| TRISTATE | I/O tristated |
| DRIVE 1 | Output static 1 |
| DRIVE 0 | Output static 0 |
| POWER# | This output is used to control power to an external logic via switch to cut power off during unconfigured USB device and USB suspend. 0 - USB device in Configured state 1 - USB device in Unconfigured state or during USB suspend mode |
| TXLED# | Drives LED during USB transmit |
| RXLED# | Drives LED during USB receive |
| TX or RX LED# | Drives LED during USB transmit or receive |
| BCD0 BCD1 | Configurable battery charger detect pins to indicate type of USB charger (SDP, CDP, or DCP) Configuration example: 00 - Draw up to 100 mA (Unconfigured state) 01 - SDP (up to 500 mA) 10 - CDP/DCP (up to 1.5 A) 11 - Suspend (up to 2.5 mA) This truth table can be configured using the configuration utility |
| BUSDETECT | VBUS detection. Connect VBUS to this pin via resistor network for VBUS detection when using BCD feature (refer to page 20). |
| CS0, CS1, CS2, CS3, CS4, CS5, CS6, CS7 | CapSense button input (Max up to 8) |
| CSout0, CSout1, CSout2, CSout3 | Indicates which CapSense button is pressed |
| Cmod (Available on GPIO_0 only) | External modulator capacitor, connect a 2.2 nF capacitor ($\pm 10\%$) to ground |
| Cshield (optional) | Shield for waterproofing |
| Note: These signal options can be configured on any of the available GPIO pins using Cypress-supplied configuration utility. | |

USB Power Configurations

The following section describes possible USB power configurations for the CY7C65215/CY7C65215A. Refer to the [Pin Description](#) on page 15 for signal details.

USB Bus-Powered Configuration

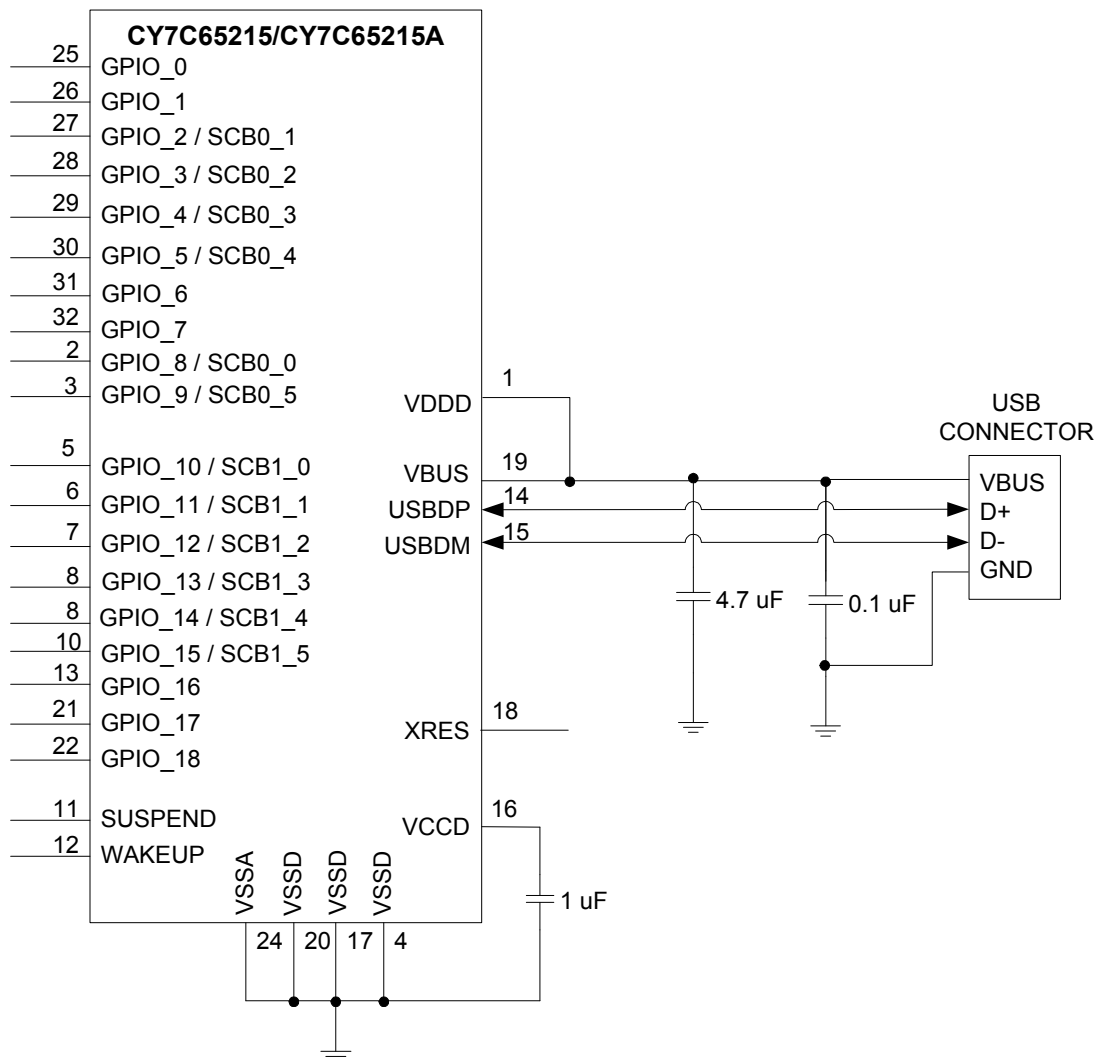
Figure 4 shows an example of the CY7C65215/CY7C65215A in a bus-powered design. VBUS is connected directly to the CY7C65215/CY7C65215A because it has an internal regulator.

The USB bus-powered system must comply with the following requirements:

1. The system should not draw more than 100 mA prior to USB enumeration (Unconfigured state).
2. The system should not draw more than 2.5 mA during USB Suspend mode.
3. A high-power bus-powered system (can draw more than 100 mA when operational) must use POWER# (configured over GPIO) to keep the current consumption below 100 mA prior to USB enumeration, and 2.5 mA during USB Suspend state.
4. The system should not draw more than 500 mA from the USB host.

The configuration descriptor in the CY7C65215/CY7C65215A flash should be updated to indicate bus power and the maximum current required by the system using the configuration utility.

Figure 4. Bus-Powered Configuration



Self-Powered Configuration

Figure 5 shows an example of CY7C65215/CY7C65215A in a self-powered design.

In this configuration:

- VBUS is powered from USB VBUS. VBUS pin is also used to detect USB connection.
- VDDD is powered from an external power supply.

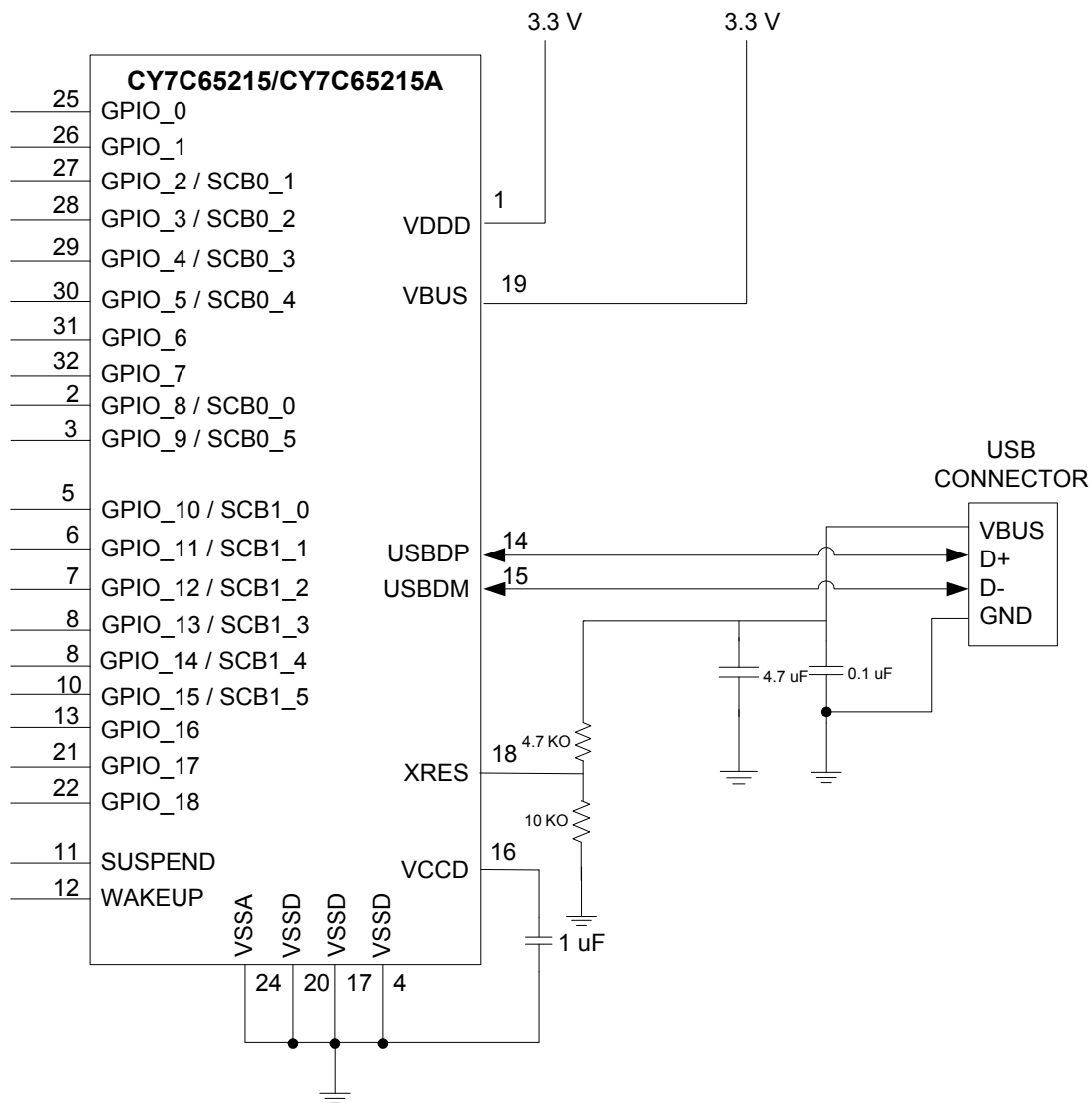
When VBUS is present, CY7C65215/CY7C65215A enables an internal, 1.5-k Ω pull-up resistor on USBDP. When VBUS is absent (USB host is powered down), CY7C65215/CY7C65215A

removes the 1.5-k Ω pull-up resistor on USBDP, and this ensures no current flows from the USBDP to the USB host via a 1.5-k Ω pull-up resistor, to comply with USB 2.0 specification.

When reset is asserted to CY7C65215, all the I/O pins are tristated.

Using the configuration utility, the configuration descriptor in the CY7C65215/CY7C65215A flash should be updated to indicate that it is self-powered.

Figure 5. Self-Powered Configuration



USB Bus Powered with Variable I/O Voltage

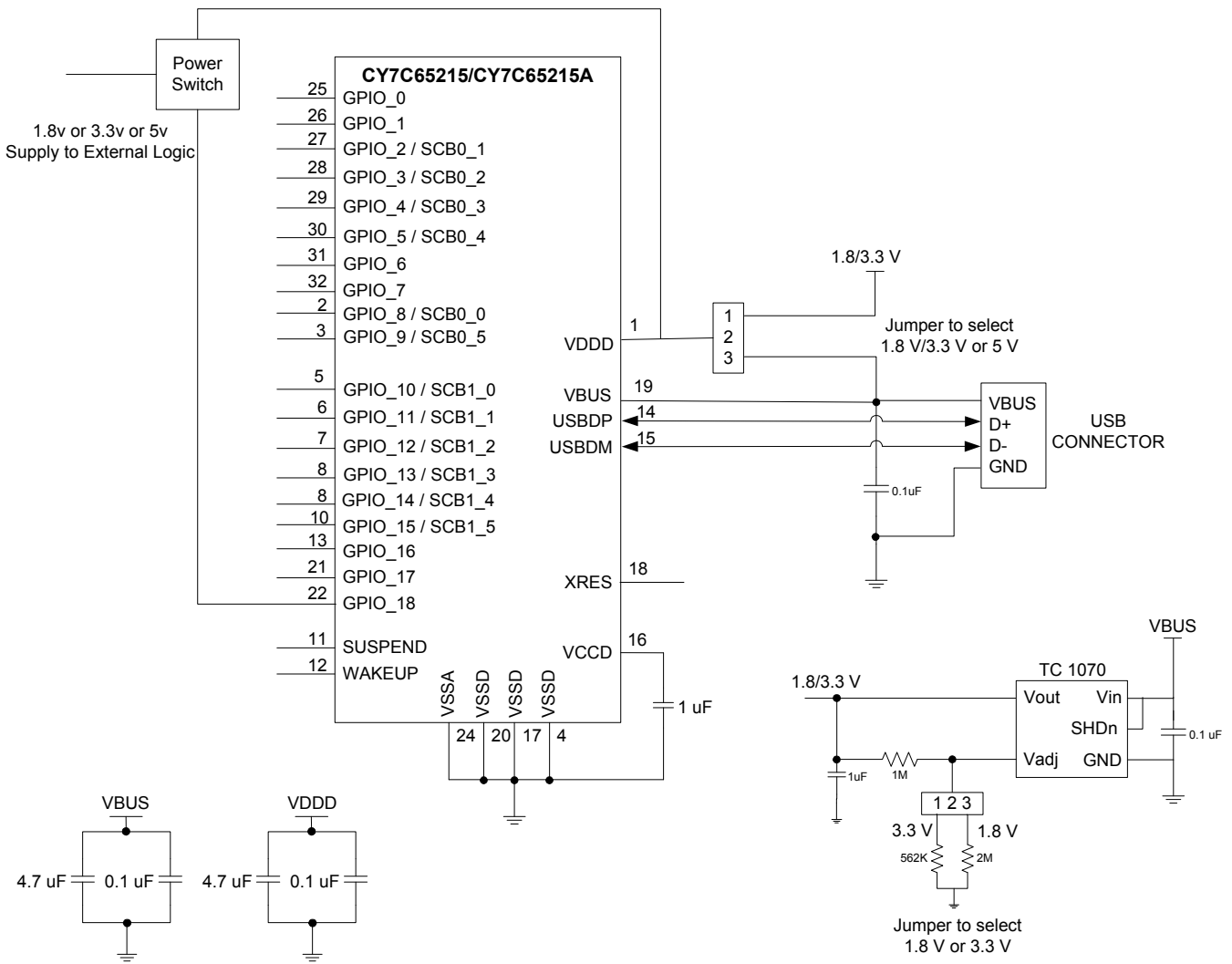
Figure 6 shows CY7C65215/CY7C65215A in a bus-powered system with variable I/O voltage. A low dropout (LDO) regulator is used to supply 1.8 V or 3.3 V (using a jumper switch) the input of which is 5 V from VBUS. Another jumper switch is used to select 1.8/3.3 V or 5 V from VBUS for the VDDD pin of CY7C65215/CY7C65215A. This allows I/O voltage and supply to external logic to be selected among 1.8 V, 3.3 V, or 5 V.

The USB bus-powered system must comply with the following:

- The system should not draw more than 100 mA prior to USB enumeration (Unconfigured state).

- The system should not draw more than 2.5 mA during USB Suspend mode.
- A high-power bus-powered system (can draw more than 100 mA when operational) must use POWER# (configured over GPIO) to keep the current consumption below 100 mA prior to USB enumeration and 2.5 mA during USB Suspend state.

Figure 6. USB Bus-Powered with 1.8 V, 3.3 V, or 5 V Variable I/O Voltage [4]



Note

4. 1.71 V ≤ VDDDD ≤ 1.89 V - Short VCCD pin with VDDDD pin; VDDDD > 2 V - connect a 1-μF decoupling capacitor to the VCCD pin.

Application Examples

The following section provides CY7C65215/CY7C65215A application examples.

USB-to-Dual UART Bridge with Battery-Charge Detection

CY7C65215/CY7C65215A can connect any embedded system, with a serial port, to a host PC through USB. CY7C65215/CY7C65215A enumerates as a dual COM port on the host PC.

SUSPEND is connected to the MCU to indicate USB suspend or USB Unconfigured and the WAKEUP pin is used to wake up CY7C65215/CY7C65215A, which in turn issues a remote wakeup to the USB host. GPIO1 and GPIO0 are configured as RXLED# and TXLED# to drive two LEDs indicating data receive and transmit respectively.

CY7C65215/CY7C65215A implements the battery charger detection functionality based on the USB Battery Charging Specification Rev 1.2.

Battery-operated bus power systems must comply with the following conditions:

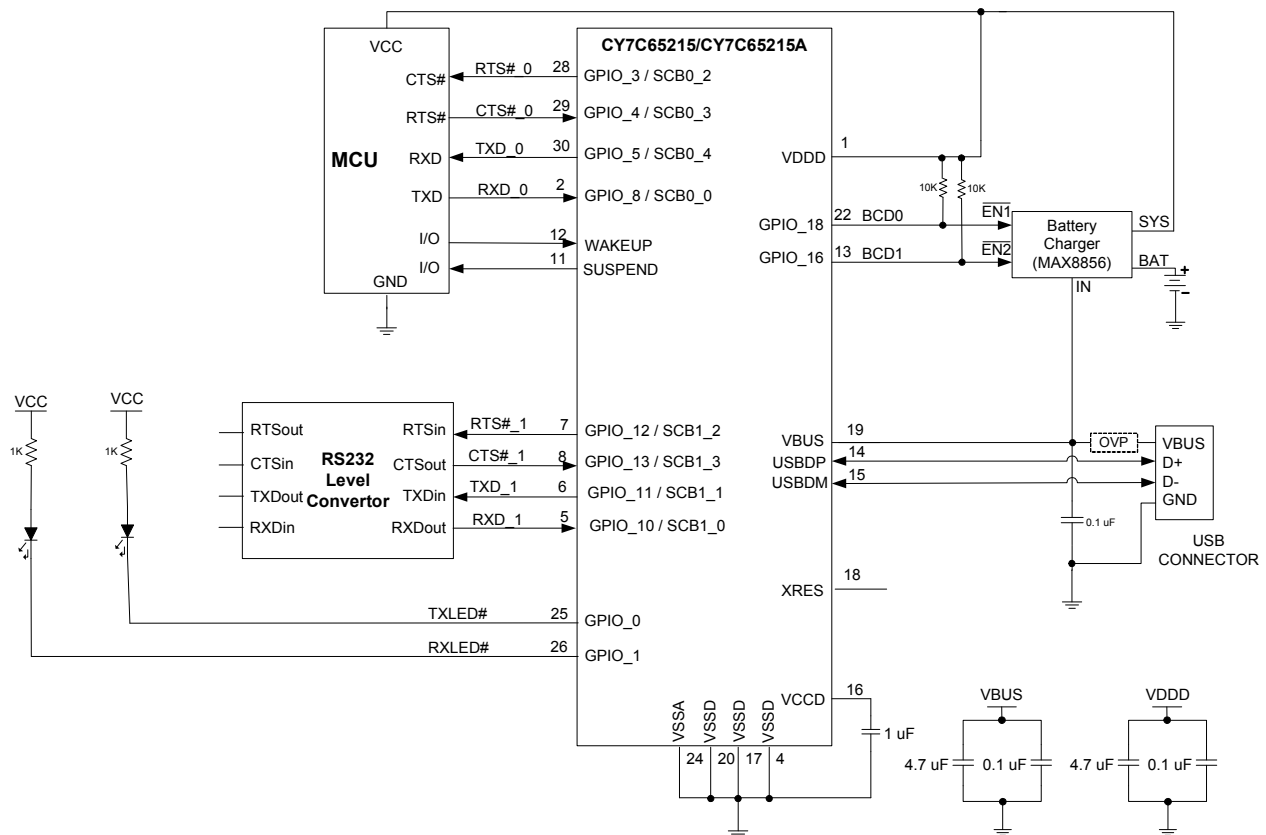
- The system can be powered from the battery (if not discharged) and be operational if VBUS is not connected or powered down.
- The system should not draw more than 100 mA from the VBUS prior to USB enumeration and USB Suspend mode.

- The system should not draw more than 500 mA for SDP and 1.5 A for CDP/DCP

To comply with the first requirement, VBUS from the USB host is connected to the battery charger as well as CY7C65215/CY7C65215A as shown in Figure 7. When VBUS is connected, CY7C65215/CY7C65215A initiates battery charger detection and indicates the type of USB charger over BCD0 and BCD1. If the USB charger is SDP or CDP, CY7C65215/CY7C65215A enables a 1.5-K pull-up resistor on the USBDP for Full-Speed enumeration. When VBUS is disconnected CY7C65215/CY7C65215A indicates absence of the USB charger over BCD0 and BCD1, and removes the 1.5-K pull-up resistor on USBDP. Removing this resistor ensures no current flows from the supply to the USB host through the USBDP, to comply with the USB 2.0 specification.

To comply with the second and third requirements, two signals (BCD0 and BCD1) are configured over GPIO to communicate the type of USB host charger and the amount of current it can draw from the battery charger. The BCD0 and BCD1 signals can be configured using the configuration utility.

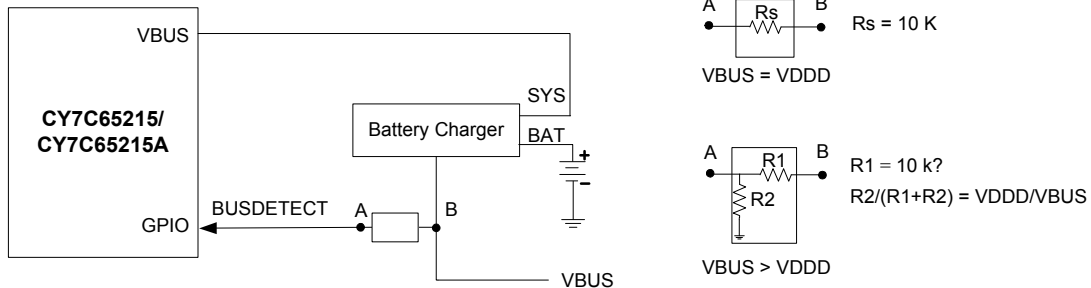
Figure 7. USB to Dual UART Bridge with Battery Charge Detection^[5]



Note

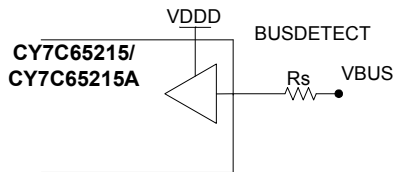
5. Add a 100 K pull-down resistor on the V_{BUS} pin for quick discharge.

In a battery charger system, a 9-V spike on the VBUS is possible. The CY7C65215/CY7C65215A VBUS pin is intolerant to voltage above 6 V. In the absence of over-voltage protection (OVP) on the VBUS line, VBUS should be connected to BUSDETECT (GPIO configured) using the resistive network and the output of battery charger to the VBUS pin of CY7C65215/CY7C65215A, as shown in the following figure.



When VBUS and VDDD are at the same voltage potential, VBUS can be connected to GPIO using a series resistor (R_s). This is shown in Figure 8. If there is a charger failure and VBUS becomes 9 V, then the 10-k Ω resistor plays two roles. It reduces the amount of current flowing into the forward biased diodes in the GPIO, and it reduces the voltage seen on the pad.

Figure 8. GPIO VBUS Detection, VBUS = VDDD



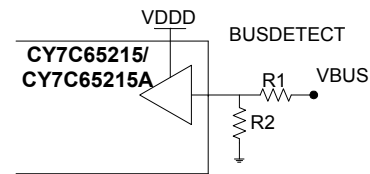
When VBUS > VDDD, a resistor voltage divider is necessary to reduce the voltage from VBUS down to VDDD for the GPIO sensing the VBUS voltage. This is shown in the following figure. The resistors should be sized as follows:

$$R_1 \geq 10 \text{ K}$$

$$R_2 / (R_1 + R_2) = V_{DDD} / V_{BUS}$$

The first condition limits the voltage and current for the charger failure situation, as described in the previous paragraph, while the second condition allows for normal-operation VBUS detection.

Figure 9. GPIO VBUS Detection, VBUS > VDDD



USB to RS485 Application

CY7C65215 can be configured as dual USB to UART interface. This UART interface operates at TTL level and it can be converted to RS485 interface using a GPIO and any half duplex RS485 transceiver IC (to convert TTL level to RS485 level) as shown in following figure. This GPIO (TXD Enable) enables and disables the RS485 transceiver IC based on availability of character in UART buffer of CY7C65215A. This

GPIO can be configured using USB-Serial Configuration utility. Figure 11 shows timing diagram of this GPIO.

RS485 is a multi-drop network – i.e. many devices can communicate with each other over a single two wire cable connection. The RS485 cable requires to be terminated at each end of the cable. Links are provided to allow the cable to be terminated if the device is physically positioned at either end of the cable.

Figure 10. USB to RS485 Bridge

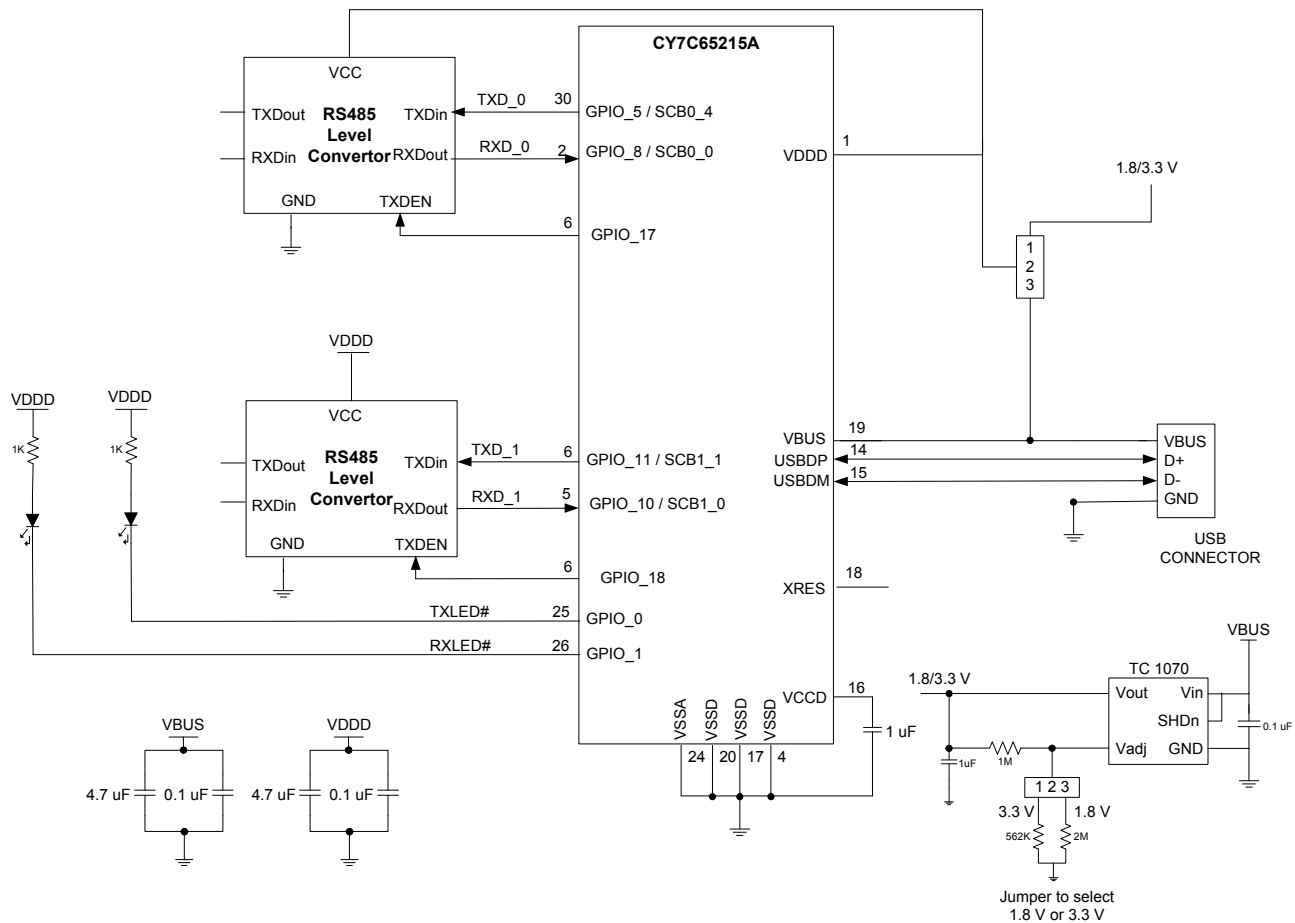
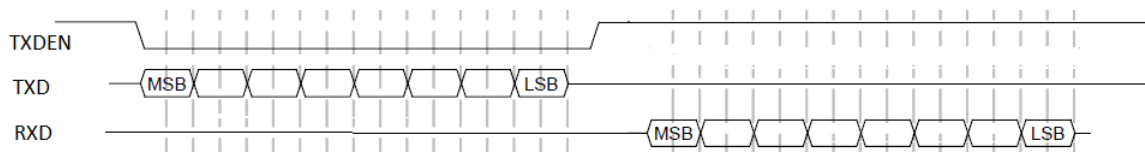


Figure 11. RS485 GPIO (TXDEN) Timing diagram



CapSense

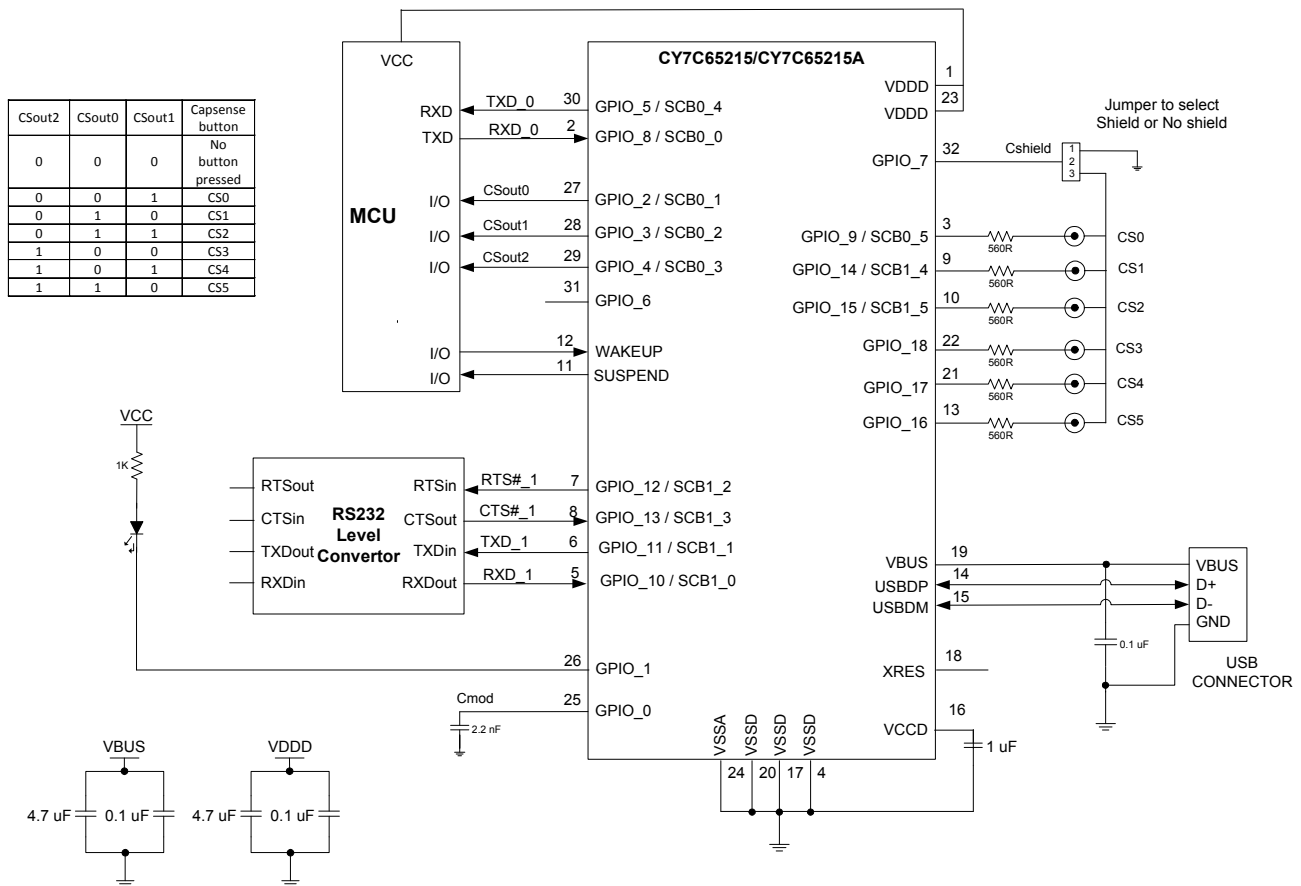
In Figure 12 CY7C65215/CY7C65215A is configured to support six CapSense buttons. Three GPIOs (CSout0, CSout1, and CSout2) are configured to indicate which CapSense button is pressed by the finger. It also implements a 2-pin UART on SCB0 and a 4-pin UART on SCB1.

A 2.2-nF (10%) capacitor (Cmod) must be connected on the GPIO_0 pin for proper CapSense operation. Optionally, the GPIO_7 pin is configured as Cshield and connected to the shield

of the CapSense (pin 2 of Watershield jumper) as shown in Figure 12. The shield prevents false triggering of buttons due to water droplets and guarantees CapSense operation (sensors respond to finger touch).

For further information on CapSense, refer to [Getting Started with CapSense](#).

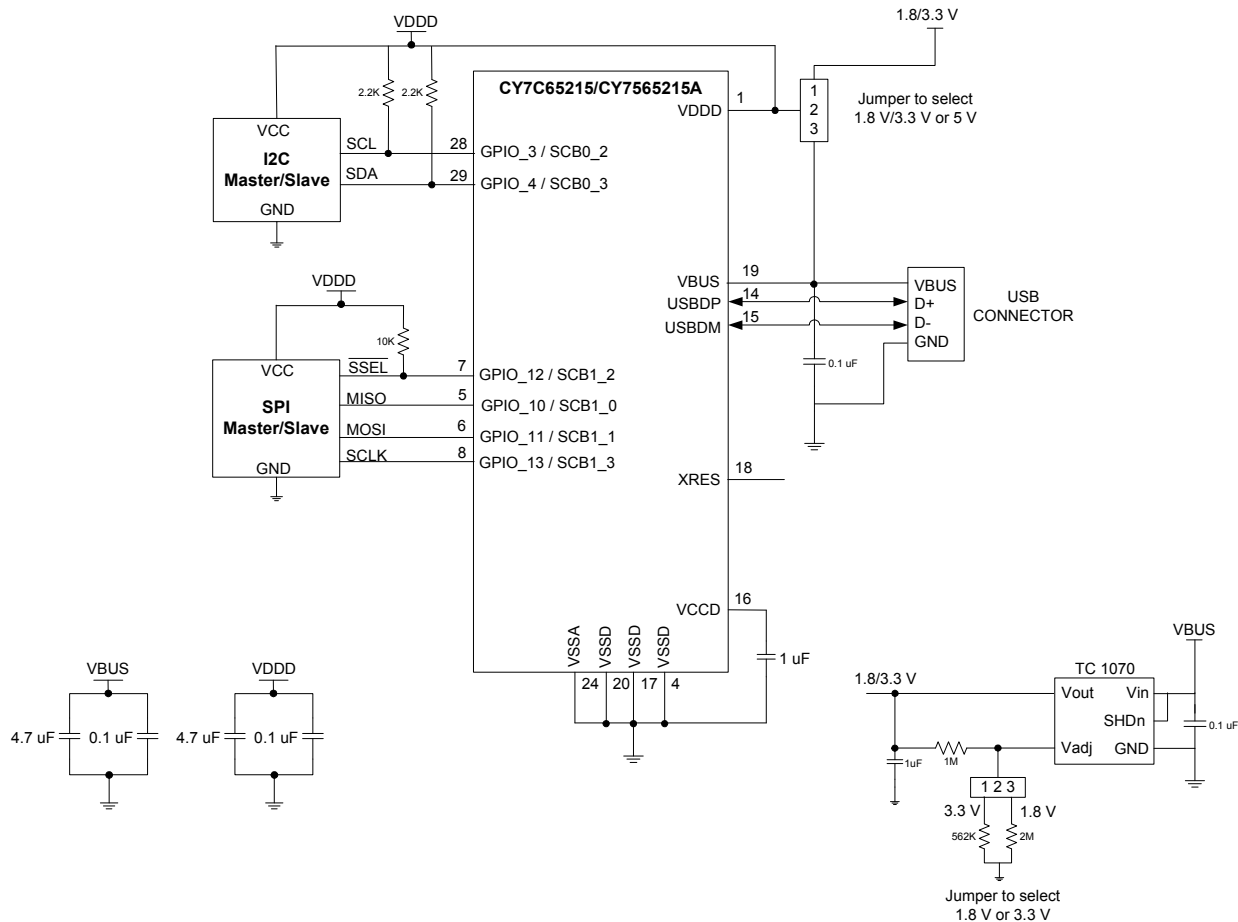
Figure 12. CapSense Schematic



USB to Dual Channel (I²C/SPI) Bridge

In Figure 13, CY7C65215/CY7C65215A is configured as a USB-to-Dual Channel (I²C/SPI) Bridge. GPIO1 and GPIO0 are configured as RXLED# and TXLED# to drive two LEDs indicating data USB receive and transmit respectively.

Figure 13. USB to I2C/SPI Bridge



I2C

The CY7C65215/CY7C65215A I2C can be configured as a Master or Slave using the configuration utility. CY7C65215/CY7C65215A supports I2C data rates up to 100 kbits/s in the standard mode (SM) and 400 kbits/s in the fast mode (FM).

In the master mode, SCL is output from CY7C65215/CY7C65215A. In the slave mode, SCL is input to CY7C65215/CY7C65215A. The I2C slave address for CY7C65215/CY7C65215A can be configured using the configuration utility. The SDA data line is bi-directional in the master and slave modes. The drive modes of the SCL and SDA port pins are always open drain.

Refer to the NXP I2C specification for further details on protocol.

SPI

The CY7C65215/CY7C65215A SPI can be configured as a Master or Slave using the configuration utility. CY7C65215/CY7C65215A supports SPI master frequency up to 3 MHz and SPI slave frequency up to 1 MHz. It can support transaction sizes ranging from 4 bits to 16 bits, which can be configured using the configuration utility.

In the master mode, SCLK, MOSI and SSEL lines act as output and MISO acts as an input. In the slave mode, SCL SCLK, MOSI, and SSEL lines act as input and MISO acts as an output.

CY7C65215/CY7C65215A supports three versions of the SPI protocol:

- Motorola - This is the original SPI protocol.
- Texas Instruments - A variation of the original SPI protocol in which data frames are identified by a pulse on the SSEL line.
- National Semiconductors - A half-duplex variation of the original SPI protocol.