Secure Digital Input/Output (SDIO) Client Controller Data Sheet

Proven System Block (PSB) for QuickLogic® Customer Specific Standard Products (CSSPs)

Features

QuickLogic CSSPs are architected from a unique combination of semiconductor solution platforms and PSBs based on customer requirements. This data sheet represents a specific PSB that is available for inclusion in a CSSP. To discuss options for adding this PSB to an existing CSSP, or architecting a new CSSP, contact your QuickLogic Customer Solution Architect (CSA).

The QuickLogic SDIO Client Controller has the following features:

- Compliant with the *SDIO Specification, Version 1.10* (not including SPI mode)
- Interrupt generation capability (the SDIO Client Controller passes interrupts from functions to the SDIO Host Controller by generating a hardware interrupt signal)
- Buffered operation with separate transmit and receive FIFO – 512-byte deep
- 1.8/2.5/3.3 V I/O voltages are supported
- SD/SDIO 1-bit and 4-bit up to 25 MHz
- CIS/CSA functionality can be implemented in an outside chip EPROM
- SDIO Commands 52 and 53
- Supports up to seven I/Os for adding functionalities such as Bluetooth®, WiFi, modem, GPS, camera, scanner, and 802.11b/g

Overview

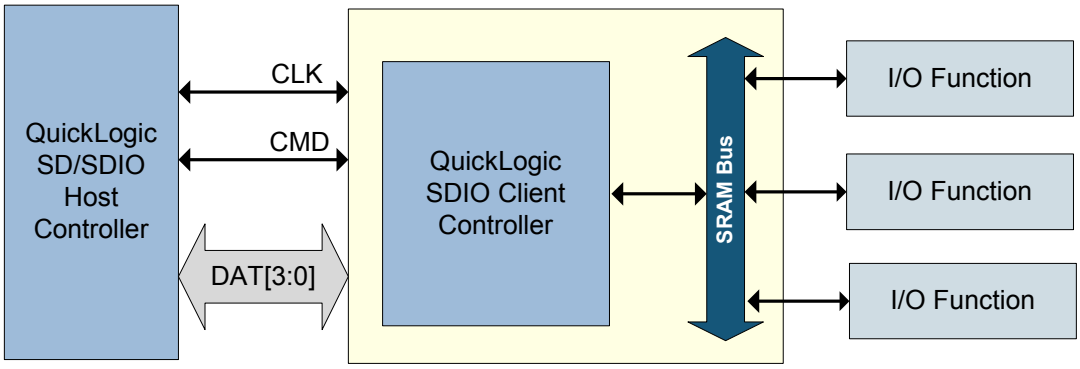
SD/SDIO interface has become one of the standards for high speed storage and peripheral connectivity interface. It is widely supported by most baseband and application processors.

The QuickLogic SDIO Client Controller PSB solution provides support for connecting the SDIO client to an SD/SDIO Host Controller. It supports several different types of functionality for connected devices such as SD cards, WiFi and mobile TV. It provides the standard SD/SDIO as the upstream interface and generic SRAM downstream interface. Taking advantage of the flexibility offered by QuickLogic's patented Vialink® technology, the downstream port can be easily adapted to accommodate different peripheral connectivity requirements. It is particularly powerful for building multi-functional SD cards. For example, a customer can design a combo SD card with a mobile TV chipset and flash memory. The built-in storage can be used for application and ESG/EPG software.

SDIO Client Controller Architecture

Figure 2 shows the basic block diagram of the SDIO Client Controller architecture.

Figure 1: SDIO Client Controller Architecture Block Diagram



Interface List and Description

Table 1 summarizes the SDIO Client Controller interface signals.

Table 1: SDIO Client Controller Interface

Signal Name	Direction	Description
SD/SDIO Signals		
SDDAT[3]	Input/Output	Data line 3
CMD	Input/Output	Command
GND		GND
GND		GND
VDD		VDD
SDCLK	Input	SDIO clock
SDDAT[0]	Input/Output	Data line 0
SDDAT[1]	Input/Output	Data line 1
SDDAT[2]	Input/Output	Data line 2
SRAM Signals		
Ext_clk	Input	SRAM bus interface clock
SRAM_CSn (1 ~ 7)	Output	SRAM bus chip select (up to 7)
SRAM _WEn	Output	SRAM bus write enable
SRAM_OEn	Output	SRAM bus output enable
SRAM_ADDR[16:0]	Output	SRAM bus addr16 ~ addr0
SRAM_DATA[7:0]	Input/Output	SRAM bus data7 ~ data0

Register Descriptions

This section describes the memory space defined in the SDIO Client Controller and the register sets:

- Card Common Control Registers (CCCR)
- Function Basic Registers (FBR)

Memory Map

Table 2 provides the memory map of the SDIO Client Controller register set and a quick reference to the register locations. Refer to the sections that follow the memory map for more details about each register.

NOTE: Registers and memory spaces shown in gray are not used, but are available for use as required.

Table 2: Memory Map

Offset Address	Register Name	Register Description
0x00_0000 ~ 0x00_00FF	CCCR	Card Common Control Registers
0x00_0100 ~ 0x00_01FF	FBR (Function 1)	Function Basic Registers (Function 1)
0x00_0200 ~ 0x00_02FF	FBR (Function 2)	Function Basic Registers (Function 2)
0x00_0300 ~ 0x00_03FF	FBR (Function 3)	Function Basic Registers (Function 3)
0x00_0400 ~ 0x00_04FF	FBR (Function 4)	Function Basic Registers (Function 4)
0x00_0500 ~ 0x00_05FF	FBR (Function 5)	Function Basic Registers (Function 5)
0x00_0600 ~ 0x00_06FF	FBR (Function 6)	Function Basic Registers (Function 6)
0x00_0700 ~ 0x00_07FF	FBR (Function7)	Function Basic Registers (Function 7)
0x00_1000 ~ 0x00_7FFF	CIS Area	Card Information Structures (Common and per-function)

Card Common Control Registers (CCCR)

The CCCR is defined in the *SDIO Specification, Version 1.10*. The CCCR is used to manage the capability and behavior, and report the status of the SDIO Client Controller. The CCCR is located at function 0 from 0x00 to 0xFF. All register bits are zero by default unless designated otherwise.

NOTE: The register bits shown in gray are Read-only or reserved for future use (RFU).

Table 3: CCCR Registers

Address	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00	CCCR/SDIO Revision	0	0	0	1	0	0	0	1
0x01	SD Spec. Revision	RFU	RFU	RFU	RFU	0	0	0	1
0x02	I/O Enable	IOE7	IOE6	IOE5	IOE4	IOE3	IOE2	IOE1	RFU
0x03	I/O Ready	IOR7	IOR6	IOR5	IOR4	IOR3	IOR2	IOR1	RFU
0x04	Int Enable	IEN7	IEN6	IEN5	IEN4	IEN3	IEN2	IEN1	IEN7
0x05	Int Pending	INT7	INT6	INT5	INT4	INT3	INT2	INT1	RFU
0x06	I/O Abort	RFU	RFU	RFU	RFU	RES	AS2	AS1	AS0
0x07	Bus Interface Control	CD disable	1	ECSI	RFU	RFU	RFU	Bus wl	Bus w0

Table 3: CCCR Registers *(Continued)*

Address	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x08	Card Capability	0	0	E4MI	1	0	0	1	0
0x09 ~ 0x0B	Common CIS Pointer	N/A							
0x0C	Bus Suspend	RFU	RFU	RFU	RFU	RFU	RFU	0	0
0x0D	Function Select	0	RFU	RFU	RFU	0	0	0	0
0x0E	Exec Flags	0	0	0	0	0	0	0	0
0x0F	Ready Flags	0	0	0	0	0	0	0	0
0x10 ~ 0x11	FN0 Block Size	I/O block size for Function 0 (0x0004)							
0x12	Power Control	RFU						EMP C	1
0x13 ~ 0xEF	RFU	RFU							
0xF0 ~ 0xFF	Reserved for Vendors								

Function Basic Registers (FBR)

Each supported I/O function has a 256-byte area FBR in addition to the CCCR. The address of this 256-byte area ranges from 0x00n00 to 0x00nFF, where n is the function number (1 through 7). The FBR defines the capability and requirements of each function.

NOTE: Registers and memory spaces shown in gray are reserved, Read-only, or not used (but are available for use as required).

Table 4: Function Basic Registers (Functions 1 through 7)

Address	7	6	5	4	3	2	1	0
0xn00	Function n CSA enable	Function n supports CSA	RFU	RFU	Function n Standard SDIO Function interface code			
0xn01	Function n Extended standard SDIO Function interface code							
0xn02	RFU	RFU	RFU	RFU	RFU	RFU	EPS	SPS
0xn03~ 0xn08	RFU							
0xn09~ 0xn0B	Pointer to Function n Card Information Structure (CIS)							
0xn0C~ 0xn0E	Pointer to Function n Code Storage Area (CSA)-N/A							
0xn0F	Data access window to Function n Code Storage Area (CSA)-N/A							
0xn10~ 0xn11	I/O block size for Function n							
0xn12~ 0xn1FF	RFU							
0x800~ 0xFFF	RFU							

Card Information Structure (CIS)

The CIS provides detailed information about the card and its individual functions. The CIS is located at address space 0x000100 - 0x017FFF and is implemented in the Programmable Fabric or EPROM outside of the chip.

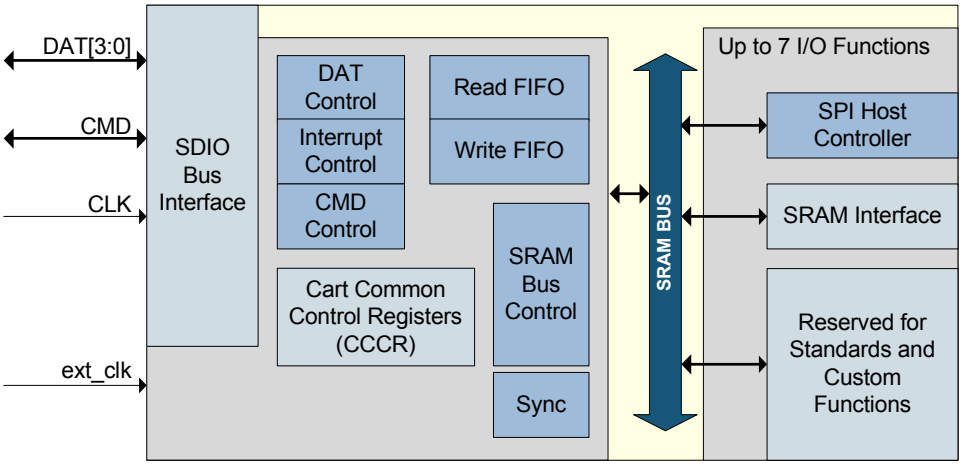
Functional and Module Description

The SDIO Client Controller provides a high-speed SD/SDIO/MMC interface to a processor and consists of the following components:

- SDIO Bus Interface
- SRAM Bus Interface
- Interrupt Control
- CMD Control
- DAT Control
- Synchronization Block
- Data FIFO

Figure 2 illustrates the SDIO Client Controller sub-module block diagram.

Figure 2: SDIO Client Controller Sub-Module Block Diagram



SDIO Bus Interface

The SDIO bus interface communicates with the SDIO Host Controller. The SDIO bus interface is responsible for decoding the address of the transfer, and directing the transaction to the CCCRs or I/O functions.

The host can initiate SDIO accesses via the SDIO bus interface module. Depending on the decoded command, this module activates and completes the initialization, command/response, and data transaction.

SRAM Bus Interface

The SRAM bus interface provides a common interface for connecting I/O functions.

When the host on the SDIO bus tries to write to I/O functions, the SRAM bus interface module is activated to pop data from the Write FIFO of the SDIO Client Controller into the registers/memory space of the I/O functions.

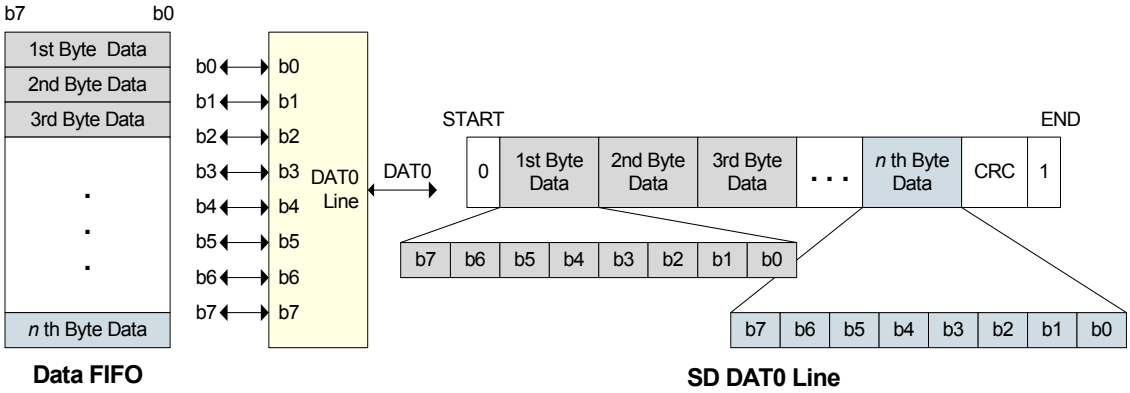
When the host on the SDIO bus tries to read from I/O functions, the SRAM bus interface module is activated to push data into the Read FIFO of the SDIO client controller from the I/O functions spaces. For this type of transfer, the SRAM bus interface continues reading from the I/O functions and pushing data into the Read FIFO until the FIFO is full or the transaction ends.

DAT Control

DAT Control is responsible for transferring data between the Data FIFO and the SD data bus, and for detecting the SDIO interrupt. It consists of an SD data multiplexer, a state machine, and DAT Line sub-modules.

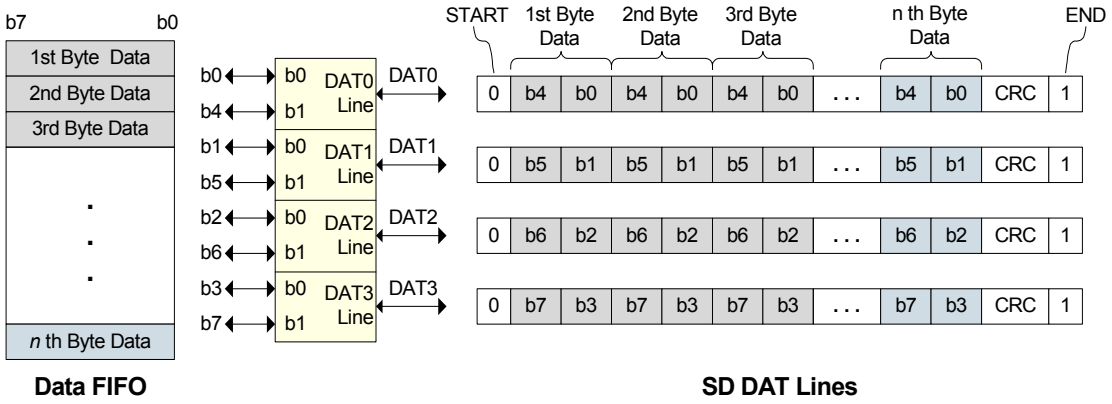
The SD data multiplexer supports 1-bit and 4-bit SD/SDIO modes. In 1-bit mode, only DAT0 is used. All 8-bit data from the Data FIFO connects to the DAT0 Line sub-module in the following way: bit0 from the Data FIFO connects to bit0 of the DAT0 Line module, bit1 from the Data FIFO connects to bit1 of the DAT0 Line module, etc. The data bit sequence on the SD data line is shown in **Figure 3**.

Figure 3: 1-bit SD Data Mode



In 4-bit mode, all four SD data lines DAT [3:0] are used. **Figure 4** shows data connections between the Data FIFO and four DAT Line modules in 4-bit mode. The data sequence in 4-bit mode is also shown.

Figure 4: 4-bit SD Data Mode



The DAT Control state machine instructs the DAT Line sub-modules to send data and generate the CRC, or to receive data and verify the CRC. It determines the end of the data packet based on the block length counter. It also detects timeout condition during busy state or waiting for read data.

Whenever the `stop` (or an `I/O abort`) command is issued, the DAT Control state machine goes to IDLE state immediately.

The SDIO interrupt function is implemented in the DAT Control module. In 1-bit SD mode, the DAT1 line is assigned as a dedicated interrupt pin. It is active low. In 4-bit mode, the SDIO interrupt logic detects the interrupt event on the DAT1 line during the interrupt period. For the definition of the interrupt period, refer to *SDIO Specification, Version 1.10*.

Interrupt Control

The SDIO Client Controller passes interrupts from functions to the SDIO Host Controller by generating a hardware interrupt signal. Each function within the SDIO Client Controller can implement interrupts as needed and is typically level sensitive. The functions' interrupt should be released only when the cause of interrupt is removed or commanded by the SDIO Host Controller. The SDIO Client Controller interrupt line is shared by multiple interrupt sources.

- 1-bit mode: Interrupt is asserted on SDDAT[1].
- 4-bit mode: Interrupt is asserted on SDDAT[1] only during a specific time or interrupt period.

CMD Control

CMD Control is responsible for receiving commands, analyzing and executing commands, and sending responses. Also, it has a Cyclic Redundancy Code (CRC) generator that checks the CRC after receiving commands and sends a CRC following a response.

To receive a command, the CMD Control state machine performs in the following manner:

1. The CMD Control state machine goes to IDLE at power up.
2. After the SD Card is initialized by the SDIO Host Controller, the CMD Control state machine goes to in Standby.

3. The CMD Control state machine monitors the CMD line and after receiving a command, checks CRC7, and then executes and sends back a response on the CMD line.
4. If the command is:
   - A Write command, the CMD Control enables the DAT Control state machine to receive data bits on DAT line(s).
   - A Read command, the CMD Control enables the DAT Control state machine to send data bits on DAT line(s) when the data is ready, followed by CRC16.
5. After completing the command, the CMD Control module releases the CMD and DAT lines and goes back to Standby.

Synchronization Block

The Sync block is used to synchronize the signals across different clock domains. It also provides a pulse signal based on the source T-flop signal's transition.

Data FIFO

The FIFO modules inside the SDIO Client Controller function as data buffers with status signals indicating the FIFO's status. There are two asynchronous 512x8-bit data FIFOs:

- TX_FIFO—for buffering transmit data
- RX_FIFO—for receiving data

When the TX_FIFO is empty, *Buffer Write Enable* is asserted to indicate that the TX_FIFO is ready to accept the next block of data.

When the RX_FIFO receives a block of data from the card, *Buffer Read Enable* is asserted to indicate to the processor that data is ready for pickup.

SDIO Commands

The SDIO Client Controller supports the following commands:

- IO_SEND_OP_COND (CMD5)
- IO_RW_DIRECT (CMD52)
- IO_RW_EXTENDED (CMD53)

Table 5 through **Table 7** shows the format and details of commands 5, 52 and 53. .

Table 5: CMD 5: IO_SEND_OP_COND

S	D	Command Index	Stuff	I/O OCR	CRC 7	E
0	1	000101	x00			1
1	1	6	8	24	7	1

Table 6: CMD 52 IO_RW_DIRECT

S	D	Command Index	R/W flag	Function Number	R/W flag	-	Register Address	-	Write Data or Stuff Bits	CRC 7	E
0	1	000101	x00			0		0			1
1	1	6	8	3	1	1	17	1	8	7	1

Table 7: CMD 53 IO_RW_EXTENDED

S	D	Command Index	R/W flag	Function Number	Block Mode	OP Code	Register Address	Byte/Block Count	CRC 7	E
0	1	110101				0				1
1	1	6	1	3	1	1	17	9	7	1

Power Consumption

The estimated power consumption of the SDIO Client Controller with SRAM interface is 109.46 mW when operating at 33 MHz (SD_CLK at 25 MHz) with supply voltages of 1.8 V (core) and 3.3 V (I/Os).

Supported Operating Systems

The SDIO Client Controller PSB supports the following operating systems:

- Windows® CE
- Windows Mobile®
- Linux®
- Android®

Contact Information

Phone: (408) 990-4000 (US)
(647) 367-1014 (Canada)
+(44) 1932-21-3160 (Europe)
+(886) 2-2345-5600 (Taiwan)
+(86) 21-5116-0532 (China)
E-mail: info@quicklogic.com
Sales: America-sales@quicklogic.com
Europe-sales@quicklogic.com
Asia-sales@quicklogic.com
Japan-sales@quicklogic.com
Support: www.quicklogic.com/support
Internet: www.quicklogic.com

Revision History

Revision	Date	Originator and Comments
A	May 2008	First release.
B	July 2010	Kathleen Bylsma

Notice of Disclaimer

QuickLogic is providing this design, product or intellectual property "as is." By providing the design, product or intellectual property as one possible implementation of your desired system-level feature, application, or standard, QuickLogic makes no representation that this implementation is free from any claims of infringement and any implied warranties of merchantability or fitness for a particular purpose. You are responsible for obtaining any rights you may require for your system implementation. QuickLogic shall not be liable for any damages arising out of or in connection with the use of the design, product or intellectual property including liability for lost profit, business interruption, or any other damages whatsoever. QuickLogic products are not designed for use in life-support equipment or applications that would cause a life-threatening situation if any such products failed. Do not use QuickLogic products in these types of equipment or applications.

QuickLogic does not assume any liability for errors which may appear in this document. However, QuickLogic attempts to notify customers of such errors. QuickLogic retains the right to make changes to either the documentation, specification, or product without notice. Verify with QuickLogic that you have the latest specifications before finalizing a product design.

Copyright and Trademark Information