SD Specifications
Part E7
iSDIO Simplifed Specification

Version 1.10

March 25, 2014

Technical Committee
SD Card Association

Revision History

Date	Version	Changes compared to previous issue
March 25, 2014	1.10	The first release of iSDIO Simplified Specification

To the extent this proposed specification, which is being submitted for review under the IP Policy, implements, incorporates by reference or refers to any portion of versions 1.0 or 1.01 of the SD Specifications (including Parts 1 through 4), adoption of the proposed specification shall require Members utilizing the adopted specification to obtain the appropriate licenses from the SD-3C, LLC, as required for the utilization of those portion(s) of versions 1.0 or 1.01 of the SD Specifications.
For example, implementation of the SD Specifications in a host device under versions 1.0 or 1.01 and under the adopted specification requires the execution of a SD Host Ancillary License Agreement with the SD-3C, LLC; and implementation of the SD Specifications under versions 1.0 or 1.01 and under the proposed specification in a SD Card containing any memory storage capability (other than for storage of executable code for a controller or microprocessor within the SD Card) requires the execution of a SD Memory Card License Agreement with the SD-3C, LLC.

Release of SD Simplified Specification/Addendum

The following conditions apply to the release of the SD Simplified Specification/Addendum by the SD Card Association. The Simplified Specification/Addendum is a subset of the complete version of SD Specification/Addendum which is owned by the SD Card Association.

Conditions for publication

Publisher and Copyright Holder:

SD Card Association
2400 Camino Ramon, Suite 375
San Ramon, CA 94583 USA
Telephone: +1 (925) 275-6615,
Fax: +1 (925) 886-4870
E-mail: office@sdcard.org

Notes:

This Simplified Specification/Addendum is provided on a non-confidential basis subject to the disclaimers below. Any implementation of the Simplified Specification/Addendum may require a license from the SD Card Association or other third parties.

Disclaimers:

The information contained in the Simplified Specification/Addendum is presented only as a standard Specification/Addendum for SD Cards and SD Host/Ancillary products and is provided "AS-IS" without any representations or warranties of any kind. No responsibility is assumed by the SD Card Association for any damages, any infringements of patents or other right of the SD Card Association or any third parties, which may result from its use. No license is granted by implication, estoppel or otherwise under any patent or other rights of the SD Card Association or any third party. Nothing herein shall be construed as an obligation by the SD Card Association to disclose or distribute any technical information, know-how or other confidential information to any third party.

Conventions Used in This Document

Naming Conventions

- Some terms are capitalized to distinguish their definition from their common English meaning.
- Words not capitalized retain their common English meaning.

Numbers and Number Bases

- Hexadecimal numbers are written with a lower case "h" suffix, e.g., FFFFh and 80h.
- Binary numbers are written with a lower case "b" suffix (e.g., 10b).
- Binary numbers larger than four digits are written with a space dividing each group of four digits, as in 1000 0101 0010b.
- All other numbers are decimal.

Key Words

- May: Indicates flexibility of choice with no implied recommendation or requirement.
- Shall: Indicates a mandatory requirement. Designers shall implement such mandatory requirements to ensure interchangeability and to claim conformance with the specification.
- Should: Indicates a strong recommendation but not a mandatory requirement. Designers should give strong consideration to such recommendations, but there is still a choice in implementation.

Application Notes

Some sections of this document provide guidance to the host implementers as follows:

> Application Note:
> This is an example of an application note.

Table of Contents

Table of Figures

Table of Tables

1. Introduction

1.1 Concept of iSDIO Specification

This Part E7 iSDIO (intelligent SDIO) Specification defines the register specification and the functions of the iSDIO Card. iSDIO is based on the Physical Layer Specification and the SDIO Specification. Especially, iSDIO Card is targeting the Combo Card which has not only a memory device but also one or more I/O devices. In the case of the current SDIO Combo Card, data is always transferred between the memory device and the I/O device via the host data path. For more efficient and intelligent data transfer, the iSDIO Card realizes an internal data transfer function between the memory device and the I/O device by simple instructions from the Host. Moreover, it supports a simple and higher level interface between the Host and the Card, which enables it to apply it to wide range of the Hosts.

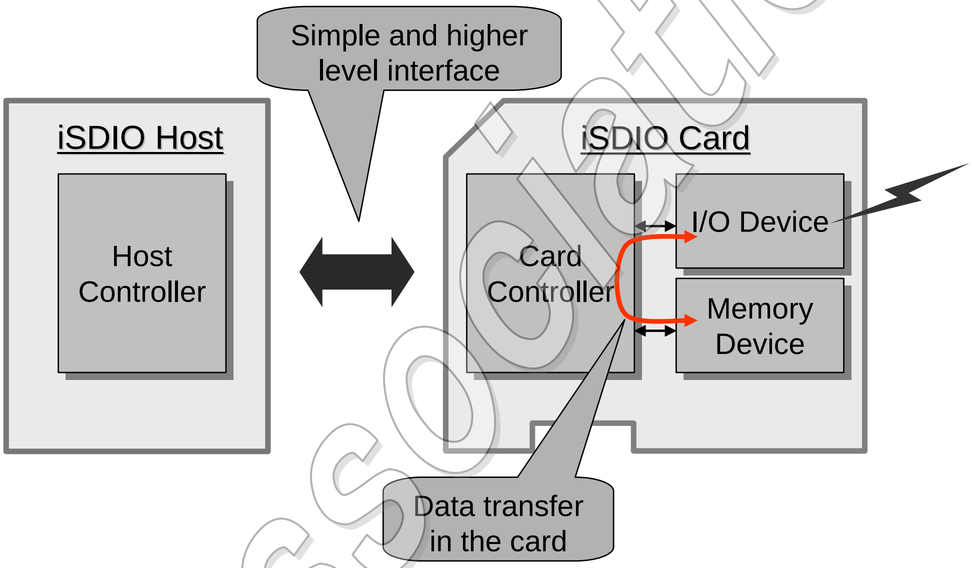

Figure 1-1 : iSDIO Concept

1.2 Typical Applications

An iSDIO Card supports I/O functions in addition to the usual memory storage functions. For example, iSDIO Wireless LAN card is a kind of Combo Card consisting of a memory device and a wireless LAN device. This iSDIO Wireless LAN card can be used as not only a legacy memory card with SD memory Host, but also can add wireless LAN communication functions to SD Host compliant with the iSDIO specifications.

Figure 1-2 provides a typical application of iSDIO Wireless LAN card. In this usage, a Host records data to an iSDIO Wireless LAN card by the same method as a legacy SD memory card. The user can select some data by using the User Interface of the Host, and upload them to Server or the other devices by using wireless LAN communication functions provided by the iSDIO Wireless LAN card. This enables them to share their data with others instantly.

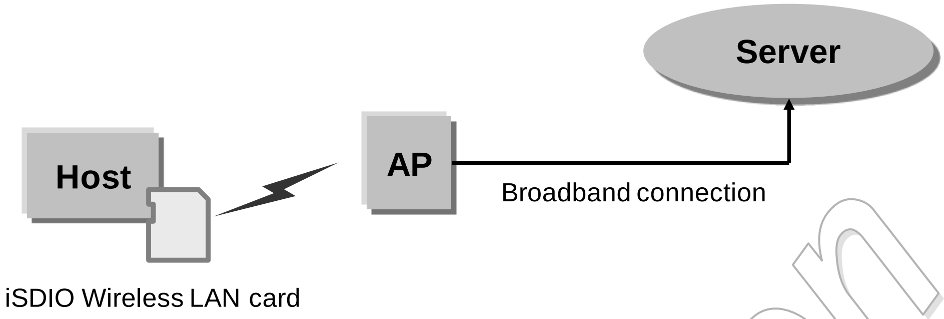

Figure 1-2 : An Example of iSDIO Application

1.3 iSDIO Specifications Structure

The iSDIO Card is a Combo Card that supports both memory and I/O functions. The iSDIO Specification is based on the Physical Layer Specification ([SDPart1]) and the SDIO Specification ([SDIO]). Therefore, the other features of iSDIO Card not specified in this iSDIO Specification are compatible with the above two base specifications. They include mechanical, electrical, power, signaling, and software features.

The iSDIO Specification consists of the following two layers, the Common Interface Layer and the Application Layer described in Figure 1-3. The Common Interface Layer defines common features among all iSDIO applications. The Application Layer defines specific features of each application. This specification defines Common Interface Layer.

There are two types of command interface for iSDIO Card. One is SD Memory Command Access by using CMD48 and CMD49 defined in [SDPart1]. The other is SDIO Command Access by using CMD52 and CMD53 defined in [SDIO]. The commands supported by each type of iSDIO Card are specified in the iSDIO Application Layer definitions.

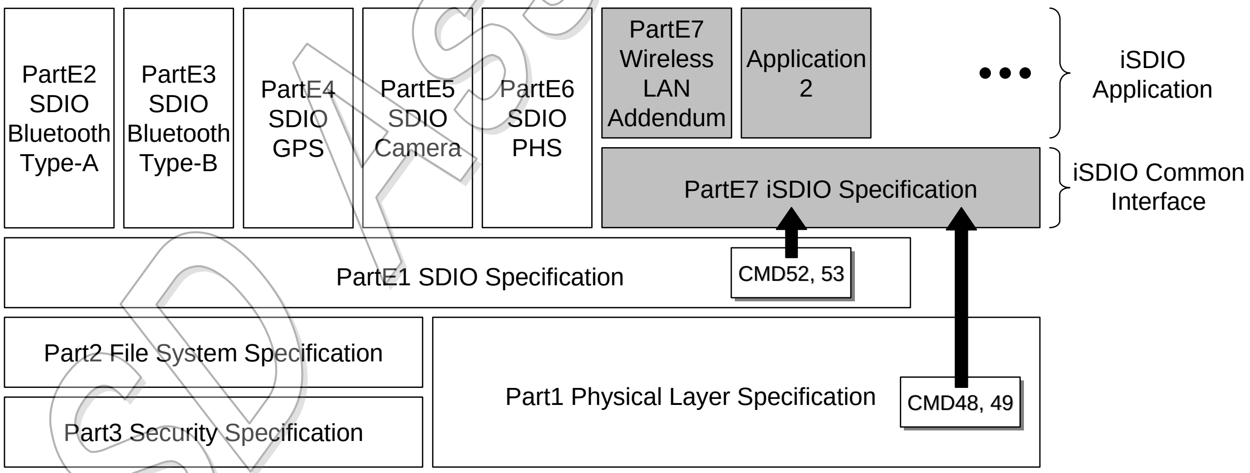
PartE2 SDIO Bluetooth Type-A
PartE3 SDIO Bluetooth Type-B
PartE4 SDIO GPS
PartE5 SDIO Camera
PartE6 SDIO PHS
PartE7 Wireless LAN Addendum
Application 2
• • •
iSDIO Application
PartE7 iSDIO Specification
iSDIO Common Interface
PartE1 SDIO Specification
CMD52, 53
Part2 File System Specification
Part3 Security Specification
Part1 Physical Layer Specification
CMD48, 49

Figure 1-3 : iSDIO Specification Structure

1.4 Endian

- iSDIO Card and Device take the **little endian** representation.
- Multiple-byte numerical values in a description field shall be recorded in the **little endian** form. e.g.) the 32-bit hexadecimal number ‘12345678h’ shall be recorded as ‘78h’, ‘56h’, ‘34h’ and ‘12h’ in this order.

2. iSDIO Card Register Definition

2.1 Register Overview

This chapter defines the iSDIO Register Map to be used for the interface between a Card and a Host. iSDIO Registers are accessed by CMD52/53 as defined in the Part E1 SDIO Specification or CMD48/49 as defined in the Part 1 Physical Layer Specification Version 4.10.

2.2 Register Definition

The iSDIO Register Space is based on the SDIO Register Map defined in Chapter **6. SDIO Card Internal Operation** of [SDIO]. This section describes additional definitions to the SDIO Register Space.

Most host devices using SD Memory Card do not support SDIO functions such as the variable block size data transfer and the interrupt service. This type of host is based on the 512-byte fixed block length data transfer. It is preferable that such a memory-based Host can use iSDIO Combo Card without supporting the full SDIO Specification. The Function Extension Specification, introduced in [SDPart1], enables memory-based host devices to use the iSDIO Combo Card. The Function Extension Specification defines CMD48/49 to access Extension Register Space. There are two types of spaces in CMD48/49; Memory Extension Register Space and I/O Extension Register Space. I/O Extension Register Space is equivalent to SDIO space. iSDIO functions are mapped to this space.

If a Host starts to access the iSDIO Register Space via CMD52/53, the Host shall continue to access via CMD52/53 and not CMD48/49. If a Host starts to access the iSDIO Register Space via CMD48/49, the Host shall continue to access via CMD48/49 and not CMD52/53.

2.2.1 CIA (Function 0) for iSDIO

CIA (function 0) for iSDIO consists of CCCR, FBRs, General Information and CIS Area. (Figure 2-1) The General Information is defined in [SDPart1] and mapped to the area from “000800h” to “000BFFh” inclusive for an iSDIO.

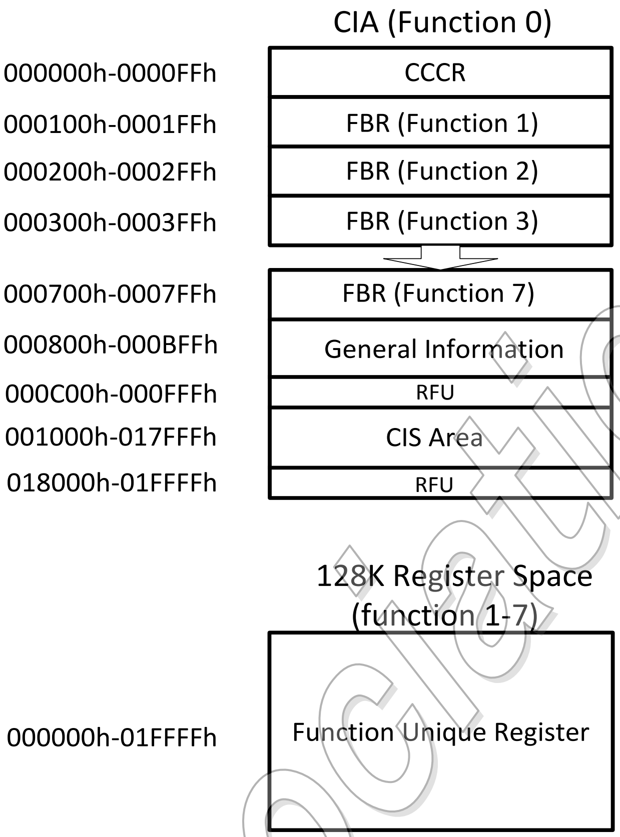

Figure 2-1 : SDIO Register Space for iSDIO

2.2.1.1 CCCR for iSDIO

This sub-section defines requirements for the support of CCCR for iSDIO.

2.2.1.1.1 Bus Settings and I/O Enable

CMD6 (Current Limit/Power Limit and/or Access Mode) influences a power consumption of a memory device. In case of an iSDIO Card, power consumption of an I/O device varies depending on the power consumption of a memory device. And this power consumption of an I/O device influences the determination process whether I/O Enable for the target I/O device is acceptable or not.
Therefore, if a Host needs to issue CMD6 (Current Limit/Power Limit and/or Access Mode), the Host shall issue the CMD6 before setting of I/O Enable to ‘1b’.

2.2.1.1.2 CCCR Support in SD Memory Command Access

In case of the SD Memory Command Access defined in [SDPart1], a Card shall implement the “CCCR/SDIO Revision”, “I/O Enable”, “I/O Ready”, “I/O Abort” and “Power Control” (excluding the grayed cells in Table 2-1) at a minimum. If other registers are implemented, appropriate values shall be set. Table 2-1 shows the values if they are fixed.
As for “I/O Ready”, the timeout value within which the function becomes ready shall be 1 second and does not refer the “RPLFE_ENAVLE_TIMEOUT_VAL” tuple in CIS.

Address	Register Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
00h	CCCR/SDIO Revision	SDIO bit 3	SDIO bit 2	SDIO bit 1	SDIO bit 0	CCCR bit 3	CCCR bit 2	CCCR bit 1	CCCR bit 0
01h	SD Specification Revision	RFU	RFU	RFU	RFU	SD bit 3	SD bit 2	SD bit 1	SD bit 0
02h	I/O Enable	IOE7	IOE6	IOE5	IOE4	IOE3	IOE2	IOE1	RFU
03h	I/O Ready	IOR7	IOR6	IOR5	IOR4	IOR3	IOR2	IOR1	RFU
04h	Int Enable	IEN7 = 0b	IEN6 = 0b	IEN5 = 0b	IEN4 = 0b	IEN3 = 0b	IEN2 = 0b	IEN1 = 0b	IENM = 0b
05h	Int Pending	INT7 = 0b	INT6 = 0b	INT5 = 0b	INT4 = 0b	INT3 = 0b	INT2 = 0b	INT1 = 0b	RFU
06h	I/O Abort	RFU	RFU	RFU	RFU	RES	AS2	AS1	AS0
07h	Bus Interface Control	CD Disable	SCSI = 0b	ECSI = 0b	RFU	RFU	S8B = 0b	Bus Width1	Bus Width0
08h	Card Capability	4BLS = 0b	LSC = 0b	E4MI = 0b	S4MI = 0b	SBS = 0b	SRW = 0b	SMB = 0b	SDC = 0b
09h -0Bh	Common CIS Pointer	0b	0b	0b	0b	0b	0b	0b	0b
0Ch	Bus Suspend	RFU	RFU	RFU	RFU	RFU	RFU	BR = 0b	BS = 0b
0Dh	Function Select	DF=0b	RFU	RFU	RFU	FS3 = 0b	FS2 = 0b	FS1 = 0b	FS0 = 0b
0Eh	Exec Flags	EX7 = 0b	EX6 = 0b	EX5 = 0b	EX4 = 0b	EX3 = 0b	EX2 = 0b	EX1 = 0b	EX0 = 0b
0Fh	Ready Flags	RF7 = 0b	RF6 = 0b	RF5 = 0b	RF4 = 0b	RF3 = 0b	RF2 = 0b	RF1 = 0b	RF0 = 0b
10h -11h	FN0 Block Size	0b	0b	0b	0b	0b	0b	0b	0b
12h	Power Control	RFU	RFU	RFU	TCP bit 2	TCP bit 1	TCP bit 0	EMPC	SMPC
13h	Bus Speed Select	RFU	RFU	RFU	RFU	BSS2	BSS1	BSS0	SHS
14h	UHS-I Support	RFU	RFU	RFU	RFU	RFU	SDDR 50	SSDR 104	SSDR 50
15h	Driver Strength	RFU	RFU	DTS1	DTS0	RFU	SDTD	SDTC	SDTA
16h	Interrupt Extension	RFU	RFU	RFU	RFU	RFU	RFU	EAI = 0b	SAI = 0b
17h -EFh	RFU	Reserved for Future Use (RFU)							
F0h -FFh	Reserved for Vendors	Area Reserved for Vendor Unique Registers							

Table 2-1 : CCCR Register Map in SD Memory Command Access

2.2.1.1.3 CCCR Support in SDIO Command Access

In case of the SDIO Command Access by using CMD52 and CMD53 defined in [SDIO], CCCR shall comply with the definitions in [SDIO].

2.2.1.2 FBR for iSDIO

This sub-section defines requirements for the support of FBR for iSDIO.

2.2.1.2.1 Additional Definitions in FBR for iSDIO

In addition to the definition in [SDIO], the followings registers are defined additionally for iSDIO.

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
100h	Function CSA enable	Function supports CSA	RFU	RFU	**Standard SDIO Function Interface Code**			
103h	**Standard iSDIO Function Interface Code**							
104h -105h	**MID_MANF SDIO Card Manufacturer Code**							
106h -107h	**MID_CARD Manufacturer Information**							
108h	**iSDIO Type Support Code**							

Table 2-2 : Additional Definitions in FBR for iSDIO

Standard SDIO Function Interface Code:

1110b: iSDIO

Standard iSDIO Function Interface Code is an 8-bit R/O register to indicate the Standard iSDIO Function Interface Code and this field is valid only if “Standard SDIO Function Interface Code” is ‘1110b’.

0000 0000b: No Standard iSDIO Function
0000 0001b: Wireless LAN

The values not defined above are reserved.

MID_MANF SDIO Card Manufacturer Code is a 2-byte R/O register.
The MID_MANF field identifies the SDIO Card's manufacturer. A two-byte code (2nd byte as non-zero) is assigned by the USB Implementers Forum (USB-IF). The code with 2nd byte zero is reserved for manufacturers who have an eight-bit JEDEC manufacturer code assigned by JEDEC Publication 106. Manufacturers may use their eight-bit JEDEC manufacturer code as the 1st byte of MID_MANF. For example, if a JEDEC manufacturer code is 89h, its MID_MANF is 0089h. USB-IF has all responsibility for managing manufacturer code including 2 byte codes which are assigned by the former PCMCIA organization. Two byte manufacturer codes assigned by USB-IF and by PCMCIA can be used for MID_MANF. If a manufacturer does not currently have a MID_MANF assigned, they should request assignment of a new manufacture code from USB-IF by mailing to admin@usb.org.
The value is stored in little endian. e.g. if the value is “0001h”, “01h” is stored in the address “104h” in the FBR and “00h” is stored in the address “105h” in the FBR.

MID_CARD Manufacturer Information is a 2-byte R/O register and is reserved for vendor unique information such as Part Number and/or Revision. The value is stored in little endian. e.g. if the value is “0001h”, “01h” is stored in the address “106h” in the FBR and “00h” is stored in the address “107h” in the FBR. If the CIS is implemented in a Card, MID_CARD Manufacturer Information contains the string corresponding to “TPLMID_CARD manufacturer information” in “CISTPL_MANFID” in the CIS.

iSDIO Type Support Code is an 8-bit R/O register to indicate the supported Card Type and this field is valid only if “Standard SDIO Function Interface Code” is ‘1110b’. The detailed definition of this field is defined in each application specification.

2.2.1.2.2 FBR Support in SD Memory Command Access

In case of SD Memory Command Access, FBR shall implement the “Standard SDIO Function Interface Code”, “Standard iSDIO Function Interface Code”, “MID_MANF SDIO Card Manufacturer Code”, “MID_CARD Manufacturer Information” and “iSDIO Type Support Code” at a minimum. Table 2-3 shows the values if they are fixed.

Address	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
100h	Function CSA enable =0b	Function supports CSA =0b	RFU	RFU	**Standard SDIO Function Interface Code** =1110b			
101h	Extended Standard SDIO Function Interface Code							
102h	PS3	PS2	PS1	PS0	RFU	RFU	EPS	SPS
103h	**Standard iSDIO Function Interface Code**							
104h -105h	**MID_MANF SDIO Card Manufacturer Code**							
106h -107h	**MID_CARD Manufacturer Information**							
108h	**iSDIO Type Support Code**							
109h -10Bh	Address Pointer to Function Card Information Structure (CIS)							
10Ch -10Eh	Address Pointer to Function Code Storage Area(CSA) = 00 00 00h							
10Fh	Data access window to Function Code Storage Area(CSA) = 00h							
110h -111h	I/O block size for Function							
112h -1FFh	Reserved for Future Use (RFU)							
200h -7FFh	Function 2 to 7 Function Basic Information Registers (FBR)							

Table 2-3 : FBR Register Map for iSDIO in SD Memory Command Access

2.2.1.2.3 FBR Support in SDIO Command Access

In case of SDIO Command Access, FBR shall comply with the definitions in [SDIO] and the definitions in **2.2.1.2.1 Additional Definitions in FBR for iSDIO**.

2.2.1.3 General Information and CIS for iSDIO

2.2.1.3.1 General Information for iSDIO

General Information is defined in [SDPart1]. This sub-section describes the additional definitions in the General Information of iSDIO.

Address	Field Name
800h -801h	**Structure Revision**

802h -803h	**General Information Length**
804h	**Number of Extensions**
805h -80Fh	Reserved
810h -811h	**Standard Function Code**
812h -813h	**Function Capability Code**
814h -815h	**Function Manufacturer Code**
816h -825h	**Function Manufacturer Name**
826h -827h	**Particular Function Code**
828h -837h	**Function Name**
838h -839h	**Pointer to Next Extension**
83Ah	**Number of Register Sets**
83Bh	Reserved
83Ch -83Fh	**Register Set Address**

Table 2-4 : Minimum Support of General Information Register Map for iSDIO

Structure Revision (2 bytes):
This field is set to “00 00h”.

General Information Length (2 bytes):
There are no additional definitions for iSDIO. See [SDPart1].

Number of Extensions (1 byte):
There are no additional definitions for iSDIO. See [SDPart1].

Standard Function Code (2 bytes):
The Standard Function Code (810h in General Information for iSDIO) contains the “Standard SDIO Interface Code” (0100h in FBR) defined in [SDIO] and **2.2.1.2.1 Additional Definitions in FBR for iSDIO**.
The Standard Function Code (811h in General Information for the iSDIO) contains the “Standard iSDIO Interface Code” (0103h in FBR) defined in **2.2.1.2.1 Additional Definitions in FBR for iSDIO**.

Function Capability Code (2 bytes):
The Function Capability Code (812h in General Information for the iSDIO) contains the “iSDIO Type Support Code” (0108h in FBR) defined in **2.2.1.2.1 Additional Definitions in FBR for iSDIO**.
The Function Capability Code (813h in General Information for the iSDIO) shall be set to “00h”.

Function Manufacturer Code (2 bytes):
The Function Manufacturer Code contains the MID_MANF SDIO Card Manufacturer Code (0104h-0105h in FBR) defined in **2.2.1.2.1 Additional Definitions in FBR for iSDIO**.

Function Manufacturer Name (16 bytes):
There are no additional definitions for iSDIO. See [SDPart1].

Particular Function Code (2 bytes):
The Particular Function Code contains the MID_CARD Manufacturer Information (0106h-0107h in FBR) defined in **2.2.1.2.1 Additional Definitions in FBR for iSDIO**.

Function Name (16 bytes):
The Function Name contains a string corresponding to the first 16-byte of “TPLLV1_INFO” in “CISTPL_VERS_1” defined in [SDIO], and if the string length of “TPLLV1_INFO” is smaller than 16 bytes, unused field are filled with “00h”.

Pointer to Next Extension (2 bytes):
There are no additional definitions for iSDIO. See [SDPart1].

Number of Register Sets (1 byte):
There are no additional definitions for iSDIO. See [SDPart1].

Register Set Address (4 bytes):
There are no additional definitions for iSDIO. See [SDPart1].

2.2.1.3.2 CIS for iSDIO

CIS is defined in [SDIO]. There are no additional CIS definitions for iSDIO.

2.2.1.3.3 General Information and CIS Support

In case of SD Memory Command Access, a Card shall support General Information but is not required to support CIS.
In case of SDIO Command Access, a Card shall support CIS but is not required to support General Information.

2.2.1.4 Summary of CIA (Function 0) Registers Required for Each Command Access

Table 2-5 shows the summary of CIA (Function 0) Registers which shall be implemented for SD Memory Command Access and SDIO Command Access.

Card Type	CCCR	FBR	General Information	CIS
SD Memory Command Access	Mandatory (at a minimum 00h, 02h, 03h, 06h and 12h are required)	Mandatory (at a minimum 100h, 103h to 108h are required)	Mandatory (at a minimum 800h to 83Fh are required)	N/A
SDIO Command Access	Mandatory	Mandatory	Optional	Mandatory

Table 2-5 : CIA (Function 0) Registers Required for Each Command Access

2.2.2 iSDIO Register

The iSDIO Register is assigned in the one of Function Unique Register.
The iSDIO Register consists of “Command Write Register Port”, “Response Data Register Port”, “Status Register” and “Capability Register”.
“Command Write Register Port” is a write-only port and used to send iSDIO Commands to the Card. After the entire Command Write Data is sent to the Port using CMD52/53 or CMD49 by a Host the Card interprets the iSDIO Command and executes the corresponding functions.
“Response Data Register Port” is a read-only port and used to read the Response Data to the corresponding iSDIO Commands issued. After the Response Data is prepared in a Card, the Command Response Data is read from the Port using CMD52/53 or CMD48 by a Host.
“Status Register” is used to read/write Status Information of a Card. This register is accessed by CMD52/53 or CMD48/49 by a Host. Some fields are read-and-write memory area and some are read-only memory area.
“Capability Register” is read-only register to indicate the Card’s capability, and a Host should check the register using CMD52/53 or CMD48 prior to any other command.
Table 2-6 shows the structure of the iSDIO Register.

Address	Name	Short Description	Type
00000h	Command Write Register Port	Data Port to write the iSDIO Command Write Data	W/O
00001h - 001FFh	Reserved		
00200h	Response Data Register Port	Data Port to read the iSDIO Command Response Data	R/O
00201h - 03FFh	Reserved		
00400h - 005FFh	Status Register	Memory Area for iSDIO Status Register	Table 2-7
00600h - 007FFh	Capability Register	Memory Area for iSDIO Capability Register	R/O
00800h - 00FFFh	Reserved		
01000h - 01FFFh	Reserved for Vendor		
02000h - 1FFFFh	Reserved		

Table 2-6 : iSDIO Register Map

Command Write Register Port is a data port to write the iSDIO Command Write Data. A Host is able to issue iSDIO command(s) by sending the iSDIO Command Write Data to this Port. Refer to **2.2.2.2 iSDIO Command Write Data**.

Response Data Register Port is a data port to read the iSDIO Command Response Data. A Host is able to receive a response to an issued iSDIO command by reading the iSDIO Command Response Data from this port. Refer to **2.2.2.3 iSDIO Command Response Data**.

Status Register is a memory area to read/write the iSDIO Status Register. Refer to **2.2.2.1 iSDIO Status Register**.
Capability Register is a memory area to read the iSDIO Capability Register. Refer to **2.2.2.4 iSDIO Capability Register**.
Reserved fields shall be filled with “00h”.

2.2.2.1 iSDIO Status Register

Table 2-7 shows the structure of the iSDIO Status Register. A Host is able to read the latest status of a Card and is able to set parameters to the Card via the register.

Address	Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Type
0400h	Command Write Status	-	-	-	-	-	-	**CWA**	**CWU** (opt)	R/W
0401h - 041Fh	Reserved									
0420h	iSDIO Status	-	-	-	-	**ASU** (Int)	**MCU** (Int)	**ESU** (Int)	**CRU** (Int)	R/W
0421h		-	-	-	-	-	-	-	-	
0422h	iSDIO Int Enable	-	-	-	-	**ASU_ ENA**	**MCU_ ENA**	**ESU_ ENA**	**CRU_ ENA**	R/W
0423h		-	-	-	-	-	-	-	-	
0424h	Error Status	-	-	-	-	**APE**	**RRE**	**CWE**	**CRE**	R/W
0425h		-	-	-	-	-	-	-	-	
0426h	Memory Status	-	-	-	-	-	-	**FAT**	**MEX**	R/O
0427h		-	-	-	-	-	-	-	-	
0428h - 043Fh	Reserved									
0440h - 04DFh	Command Response Status Queue									R/O
04E0h - 04FFh	Reserved									
0500h - 05FFh	Application Status									Note

Table 2-7 : iSDIO Status Register Map

Command Write Status:

CWU (Command Write Update) specifies whether the command writing is finished. If the “CWN” in the Capability Register (**2.2.2.4 iSDIO Capability Register**) is ‘1b’, a Host shall set this field to ‘1b’ after the command writing is finished. After a Card has received the Command Write Data, this field is reset to ‘0b’ by the Card. Note that in case that the ‘CWN’ in the Capability Register is ‘0b’, a Host is not required to set this field to ‘1b’ and even though it is set, the Card ignores this set operation and accepts the Command Write Data.

CWA (Command Write Abort) specifies whether the command writing will be aborted. During the command writing, a Host is able to set this field to 1b to abort the command writing. After a Card accepts the abort, the Card clears the received data (a part of the Command Write Data) and resets this field to 0b.

iSDIO Status:

CRU (Command Response Update) specifies whether the Command Response Status (**2.2.2.1.1.1 Command Response Status**) is updated. When the Command Response Status is changed to “02h: Command Rejected”, “03h: Process Succeeded”, “04h: Process Terminated” or “80h-FFh: Process Failed”, this field is set to 1b. This field is able to be reset to ‘0b’ by a Host. In case of an SDIO

Command Access, an interrupt is sent from a Card on the SDIO bus when this field is changed to '1b'.

ESU (Error Status Update) specifies whether the Error Status (below) is updated. When the Error Status is changed by the Card, this field is set to '1b'. This field is able to be reset to '0b' by a Host. In case of SDIO-base Card, an interrupt is sent from a Card on SDIO bus when this field is changed to '1b'.

MCU (Media Change Update) specifies whether the Memory Status (below) is updated. When the Memory Status is changed including the case that the FAT system in a memory device is changed by a Card, this field is set to '1b'. This field is able to be reset to '0b' by a Host. In case of SDIO Command Access, an interrupt is sent from a Card on the SDIO bus when this field is changed to '1b'.

ASU (Application Status Update) specifies whether the application specific status data (which is defined in each application specification) is updated. When the application specific status data is changed, this field is set to '1b'. This field is able to be reset to '0b' by a Host. In case of SDIO Command Access, an interrupt is sent from a Card on the SDIO bus when this field is changed to '1b'.

Note that not only one but two or more status may be set to '1b' by only a single event (for example, when one command error occurs, "CRU" and "ESU/CRE" are set to '1b' or another example, when the response buffer flow occurs, "CRU" and "ESU/RRE" are set to '1b') but the interruption occurs only once.

iSDIO Int Enable:

CRU_ENA specifies whether an interruption of CRU (Command Response Update) is enabled. If this field is set to '1b', an interruption of CRU is enabled. Otherwise, no interrupt will occur. Its default value is '0b'.

ESU_ENA specifies whether an interruption of ESU (Error Status Update) is enabled. If this field is set to '1b', an interruption of ESU is enabled. Otherwise, no interrupt will occur. Its default value is '0b'.

MCU_ENA specifies whether an interruption of MCU (Media Change Update) is enabled. If this field is set to '1b', an interruption of MCU is enabled. Otherwise, no interrupt will occur. Its default value is '0b'.

ASU_ENA specifies whether an interruption of ASU (Application Status Update) is enabled. If this field is set to '1b', an interruption of ASU is enabled. Otherwise, no interrupt will occur. Its default value is '0b'.

Note that above fields are always 0b in case of SD Memory Command Access.

Error Status:

CRE (Command Response Error) specifies whether an error has occurred in the Command Response Status (**2.2.2.1.1.1 Command Response Status**). When the Command Response Status is changed to "02h: Command Rejected" or "80h-FFh: Process Failed", this field is set to '1b'. This field is able to be reset to '0b' by a Host.

CWE (Command Write Error) specifies whether an error has occurred during a command writing. (e.g. command writing buffer is overflowed or command response status queue is overflowed) When an error has occurred, this field is set to '1b'. This field is able to be reset to '0b' by a Host.

RRE (Response Receive Error) specifies whether an error has occurred during a response receiving. (e.g. response receiving buffer is overflowed) When an error has occurred, this field is set to '1b'. This field is able to be reset to '0b' by a Host.

APE (Application Specific Error) specifies whether an error has occurred in an application specific process excluding command response, a command writing and a response receiving. When an error has occurred, this field is set to '1b'. This field is able to be reset to '0b' by a Host.

Memory Status:

MEX (Memory Existence): This bit specifies whether SD memory is identified or not. One of the following values shall be set.
0b: SD memory is not identified
1b: SD memory is identified
If an iSDIO Card has a card slot for an external (micro) SD Memory Card and the external (micro) SD Memory Card is not inserted in the slot, the iSDIO Card shall not provide the iSDIO function. (See **4. Adapter-Type iSDIO Card**) Therefore, this bit is not used for insert event detection for the external (micro) SD Memory Card, but is used for remove event detection for it.

FAT (FAT System): This bit specifies whether the File System in SD memory is updated. One of the following values shall be set.
0b: Not updated
1b: Updated

Command Response Status Queue: See **2.2.2.1.1 Command Response Status Queue**.

Application Status: This area is reserved for each application specification. Note that Read/Write Type is defined in each application specification.

The values not defined above are reserved.
Reserved fields shall be filled with “00h”.

2.2.2.1.1 Command Response Status Queue

Table 2-8 shows the structure of the iSDIO Command Response Status Queue in the Status Register. The Command Response Status Queue is a queue structure. The number of Command Response Status to be registered in the queue is specified by the “Number of Command Response Status in Queue” in the Capability Register. (Command Response Status #1 is the first and Command Response Status #8 is the last)

Address	Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Type
0440h - 0453h	Command Response Status #1	See Table 2-9								R/O
0454h - 0467h	Command Response Status #2									R/O
...	...	...								...
04CCh - 04DFh	Command Response Status #8									R/O

Table 2-8 : Command Response Status Queue in the Status Register

2.2.2.1.1.1 Command Response Status

Table 2-9 shows the details of each Command Response Status in the Status Register.

Size (byte)	Name	Short Description	Type
1	Status Registration	Denotes the command status registration	R/O
1	Reserved		R/O
2	iSDIO command id	Command id corresponding to one in Command Write Data.	R/O

4	iSDIO command sequence id	Command sequence id corresponding to one in Command Data.	R/O
1	Response Status	Denote the status of a specified command processing	R/O
3	Reserved		R/O
4	Reserved for Vendor Error Status	This field is reserved for vendor error status.	R/O
4	Response Data Size	Denote the size in bytes of the Response Data	R/O

Table 2-9 : Details of Command Response Status

Status Registration specifies whether Command Response Status is registered or not. The value is an integer and one of the following values shall be set.
00h: Not registered (Default)
01h: Registered

iSDIO command id specifies the command id corresponding to the one in the iSDIO Command Write Data. This value is an integer and shall be from “00 00h” to “FF FFh” inclusive and defined in each application specification.

iSDIO command sequence id specifies the command sequence id corresponding to one in the iSDIO Command Write Data. This value is an integer and shall be from “00 00 00 00h” to “FF FF FF FFh” inclusive.

Response Status specifies whether the specified command is accepted. This value is an integer and one of the following values shall be set.
00h: Initial (Default)
01h: Command Processing
02h: Command Rejected
03h: Process Succeeded
04h: Process Terminated
80h to FFh: Process Failed where
    80h: general error
    81h: argument error
    82h: network error
    83h: file system error
    84h: buffer overflow error
    85h to BFh: reserved
    C0h to FFh: reserved for each application specification
Note that “Initial” status may be skipped depending on the timing of status registration, and that “Command Processing” status may be skipped depending on the processing speed.

Reserved for Vendor Error Status specifies the reserved fields for vendor error status.

Response Data Size specifies the size in bytes of the prepared Response Data. This value is an integer.

The values not defined above are reserved.
Reserved fields shall be filled with “00h”.

2.2.2.2 iSDIO Command Write Data

Table 2-10 shows the structure of the iSDIO Command Write Data. A Host issues one or more commands by writing the iSDIO Command Write Data via the Command Write Register Port. Issued commands are processed sequentially in the order of the iSDIO Command Write Data.

Size (byte)	Name	Short Description	Type
1	iSDIO Command Write Identifier	“01h”	W/O
1	Number of iSDIO commands	The number of iSDIO commands to be registered. (1 to 8)	W/O
2	Reserved		W/O
4	Size of iSDIO Command Write Data	Size of iSDIO Command Write Data.	W/O
4	Reserved		W/O
variable	iSDIO Command Information #1		W/O
...	...	...	W/O
variable	iSDIO Command Information #n_1		W/O
(rest)	Reserved		W/O

Table 2-10 : iSDIO Command Write Data

iSDIO Command Write Identifier specifies the identifier for the iSDIO Command Write Data. “01h” shall be set.

Number of iSDIO commands specifies the number of commands to be issued. This value is an integer and shall be from 1 to 8 inclusive. Note that it depends on a Card implementation how many of commands are accepted.

Size of iSDIO Command Write Data specifies the size of the iSDIO Command Write Data from the “iSDIO Command Write Identifier” inclusive. This value is an integer and should be equal to or less than the value specified by the “Max Size of Command Write Data” in the Capability Register.

iSDIO Command Information: See Table 2-11.

The values not defined above are reserved.
Reserved fields shall be filled with “00h”.

Table 2-11 shows the iSDIO Command Information in Command Write Data.

Size (byte)	Name	Short Description	Type
2	Reserved		W/O
2	iSDIO command id	Command ID	W/O
4	iSDIO command sequence id	Sequence ID to distinguish from other issued commands.	W/O
2	Number of Arguments	Number of Arguments for the command	W/O
2	Reserved		W/O
4	Length of Argument #1	Length of the argument for the command (L_1)	W/O
L_1	Argument #1	Argument for the command	W/O
0, 1, 2 or 3	Padding #1	Padding size is either 0 (for L_1 mod4=0), 1 (for L_1 mod4=3), 2 (for L_1 mod4=2), 3 (for L_1 mod4=1)	W/O
....	....	....	W/O
4	Length of Argument #n_2	(L_n)	W/O
L_n	Argument #n_2		W/O
0, 1, 2 or 3	Padding #n_2	Padding size is either 0 (for L_n mod4=0), 1 (for L_n mod4=3), 2 (for L_n mod4=2), 3 (for L_n mod4=1)	W/O

Table 2-11 : iSDIO Command Information in Command Write Data

iSDIO command id specifies the command id which is defined in each application specification. This value is an integer and shall be from "00 00h" to "FF FFh" inclusive.

iSDIO command sequence id specifies the sequence ID to distinguish from other issued commands, which is specified by a Host. This value is an integer and shall be from "00 00 00 00h" to "FF FF FF FFh" inclusive. The same iSDIO command sequence ids shall not exist in the Command Response Status Queue (**2.2.2.1.1 Command Response Status Queue**). That is, the iSDIO command sequence id of a processing command whose Response Status is "Command Processing" state (**2.2.2.1.1.1 Command Response Status**) in the queue shall not be specified.

Number of Arguments specifies the number of Arguments for the command. The number is defined in each command. This value is an integer.

Length of Argument specifies the length of the argument for the command. This value is an integer.

Argument specifies the argument for the command. The type of this value depends on the command. Note that in case an "Argument" has null value, the "Length of Argument" shall be "00 00 00 00h" and the size of the corresponding "Argument" and "Padding" shall be zero.

The values not defined above are reserved.
Reserved fields shall be filled with "00h".
Padding bytes shall be filled with "00h".

2.2.2.3 iSDIO Command Response Data

Table 2-12 shows the structure of the iSDIO Command Response Data. A Host receives the response to the issued command by reading the iSDIO Command Response Data via the Response Data Register Port.

Size (byte)	Name	Short Description	Type
1	iSDIO Command Response Identifier	"02h"	R/O
3	Reserved		R/O
4	Size of iSDIO Command Response Data	Size of iSDIO Command Response Data	R/O
6	Reserved		R/O
2	iSDIO command id	Command id corresponding to the one in Command Write Register.	R/O
4	iSDIO command sequence id	Command sequence id corresponding to the one in Command Write Register.	R/O
4	Size of Response Data	Size of Response Data (L_1)	R/O
L_1	Response Data	Response data to the issued commands	R/O
0, 1, 2 or 3	Padding	Padding size is either 0 (for L_1 mod4=0), 1 (for L_1 mod4=3), 2 (for L_1 mod4=2), 3 (for L_1 mod4=1)	R/O
(rest)	Reserved		R/O

Table 2-12 : iSDIO Command Response Data

iSDIO Command Response Identifier specifies the identifier for the iSDIO Command Response Data. "02h" shall be set.

Size of iSDIO Command Response Data specifies the size of the iSDIO Command Response Data from the "iSDIO Command Response Identifier" inclusive. This value is an integer and should be equal to or less than the value specified by the "Size of Command Response Data" in the Capability Register.

iSDIO command id specifies the command id corresponding to the one in Command Write Data. This value is an integer and shall be from "00 00h" to "FF FFh" inclusive and its definition is defined in each application specification.

iSDIO command sequence id specifies the command sequence id corresponding to the one in Command Write Data. This value is an integer and shall be from "00 00 00 00h" to "FF FF FF FFh" inclusive.

Size of Response Data specifies the size of response data. This value is an integer.

Response Data contains the response data by the issued command, which is defined in each application specification. This value is an integer.

The values not defined above are reserved.
Reserved fields shall be filled with "00h".
Padding bytes shall be filled with "00h".

2.2.2.4 iSDIO Capability Register

Table 2-13 shows the structure of the iSDIO Capability Register. A Host is able to determine the supported functions in a Card which are optionally defined in this specification.

Address	Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Type
0600h	iSDIO Common Specification	Major Version Number				Minor Version Number				R/O
0601h	iSDIO Application Specification	Major Version Number				Minor Version Number				R/O
0602h	Command Write Update Necessity	-	-	-	-	-	-	-	CWN	R/O
0603h	Command Response Status Queue Support	-	-	-	Number of Command Response Status in Queue					R/O
0604h - 0607h	Max Size of Command Write Data									R/O
0608h - 060Bh	Max Size of Command Response Data									R/O
060Ch - 063Fh	Reserved									R/O
0640h - 07FFh	Application Capability									R/O

Table 2-13 : iSDIO Capability Register

iSDIO Common Specification specifies the version number of the iSDIO Common Specification.
0001 0000b: iSDIO Specification Version 1.00 (this version)

iSDIO Application Specification specifies the version number of the iSDIO Application Specification.
The version number of iSDIO Application Specification is defined in each application specification.

Command Write Update Necessity:
CWN specifies whether a Host needs to set the “CWU” flag in the Status Register after the Command Write Data is written.
0b: Unnecessary
1b: Necessary

Command Response Status Queue Support:
Number of Command Response Status in Queue specifies how many commands are able to be registered in the Command Response Status Queue. This value is an integer and shall be from 1 to 8 inclusive.

Max Size of Command Write Data specifies the size in bytes of the Command Write Data is able to be accepted in the Card. This value is an integer.

Max Size of Command Response Data specifies the size in bytes of the Command Response Data is able to be created in the Card. This value is an integer.

Application Capability: This area is reserved for each application specification.

The values not defined above are reserved.
Reserved fields shall be filled with “00h”.

2.3 Basic Interface Sequence

2.3.1 Command Registration

This subsection defines the command registration sequence as described below.

(1) A Host issues one or more iSDIO commands by writing the iSDIO Command Write Data (**2.2.2.2 iSDIO Command Write Data**) via Command Write Register Port.
The iSDIO commands are assumed to be processed sequentially from the iSDIO Command Information #1 (earlier) to the iSDIO Command Information #8 (later) in the iSDIO Command Write Data.
Note that a Host should know how many commands are able to be registered initially in the Command Response Status Queue by reading the "Number of Command Response Status in Queue" in the iSDIO Capability Register. See **2.2.2.4 iSDIO Capability Register**.

(2) The iSDIO Command Write Data is written by the CMD53 or CMD49. When CMD49 is used, if the iSDIO Command Write Data is equal to or smaller than 512 bytes, the data is written by one CMD49 command issuance and if the data is larger than 512 bytes, it is necessary to be written in the unit of 512 bytes by multiple CMD49 command issuances in correct order. When CMD53 is used, the iSDIO Command Write Data is written by one CMD53 command issuance, however it is possible to be written by multiple CMD53 command issuances in correct order by dividing the data.

(3) After the finish of writing the iSDIO Command Write Data, the Host shall set the "CWU" flag in the "Command Write Status" (**2.2.2.1 iSDIO Status Register**) to '1b' if the "CWN" flag in the "Command Write Update Necessity" (**2.2.2.4 iSDIO Capability Register**) is set to '1b'. Otherwise this process is not necessary.

(4) A Card receives and parses the iSDIO Command Write Data.

(5) The Card registers issued commands to the Command Response Status Queue (**2.2.2.1.1 Command Response Status Queue**) in the Status Register. The Command Response Status Queue is able to accept the number of commands calculated by the "Number of Command Response Status in Queue" minus the number of processing commands who's Response Status are "Command Processing" state (**2.2.2.1.1.1 Command Response Status**).
If the total number of received commands doesn't exceed the acceptable number of commands calculated above, all issued commands are registered in the queue. At this time, all of the processed commands already in the queue who's Response Status are "Command Rejected", "Process Succeeded", "Process Terminated" or "Process Failed" state shall be removed from the queue to register the issued commands.
If the total numbers of received commands exceed the acceptable number of commands calculated above, the later excess commands are rejected and "CWE (Command Write Error)" in the Error Status is set to '1b'.

(6) The registered commands are assumed to be processed sequentially from the Command Response Status #1 (earlier) to the Command Response Status #8 (later) in the Command Response Status Queue, and the sequential processing doesn't stop though an error has occurred in some command in the Command Response Status Queue.

(7) When the Card finishes the command process (i.e. the Response Status defined in **2.2.2.1.1.1 Command Response Status** becomes "Command Rejected", "Process Succeeded", "Process Terminated" or "Process Failed" state), the Card is able to accept additional number of commands. If the Host issues additional commands, the Card removes all of the processed commands from the queue and registers the additional commands to the queue.

Note that a Host is able to issue only one iSDIO command in the Command Write Data (i.e. without using the queue scheme) though a Card is able to accept multiple iSDIO commands in the Command Response Status Queue, and in this case only Command Response Status #1 is used in the Command Response Status Queue in the Card.

2.3.2 Command Processing

The following sequence defines the iSDIO command processing. See also Figure 2-2 for the command response status diagram.

(1) Initially the Command Response Status is in the "Initial" state. If the Card accepts the command, the status becomes the "Command Processing" state. If rejected (for example, the Card is busy, the Card doesn't support the command, etc), the status becomes the "Command Rejected" state, and "CRU (Command Response Update)", "ESU (Error Status Update)" and "CRE (Command Response Error" in the Status Register (**2.2.2.1 iSDIO Status Register**) are set to '1b'.
(Initial state may be skipped depending on the timing of status registration)

(2) The Host confirms whether the issued command is accepted or rejected by reading the Command Response Status and the Error Status.

(3) The Card processes the issued commands sequentially in the order in the iSDIO Command Write Data if the Card accepts multiple commands.

(4) If the Card finishes the command processing successfully, the status becomes the "Process Succeeded" state, and "CRU (Command Response Update)" in the iSDIO Status is set to '1b'. If failed, for example, network error, argument error, file system error (file not found, file open/close error etc), the status becomes the "Process Failed" state, and "CRU (Command Response Update)", "ESU (Error Status Update)" and "CRE (Command Response Error)" in the Status Register are set to '1b'.
(Command Processing state may be skipped depending on the processing speed)

(5) During the command is registered in the queue, the Host may terminate the command by issuing the "Abort" command. If the Card is able to accept the termination, the status becomes the "Process Terminated" state, and "CRU (Command Response Update)" in the iSDIO Status is set to '1b'.

(6) The Host confirms whether the command processing is succeeded or failed by reading the Command Response Status.

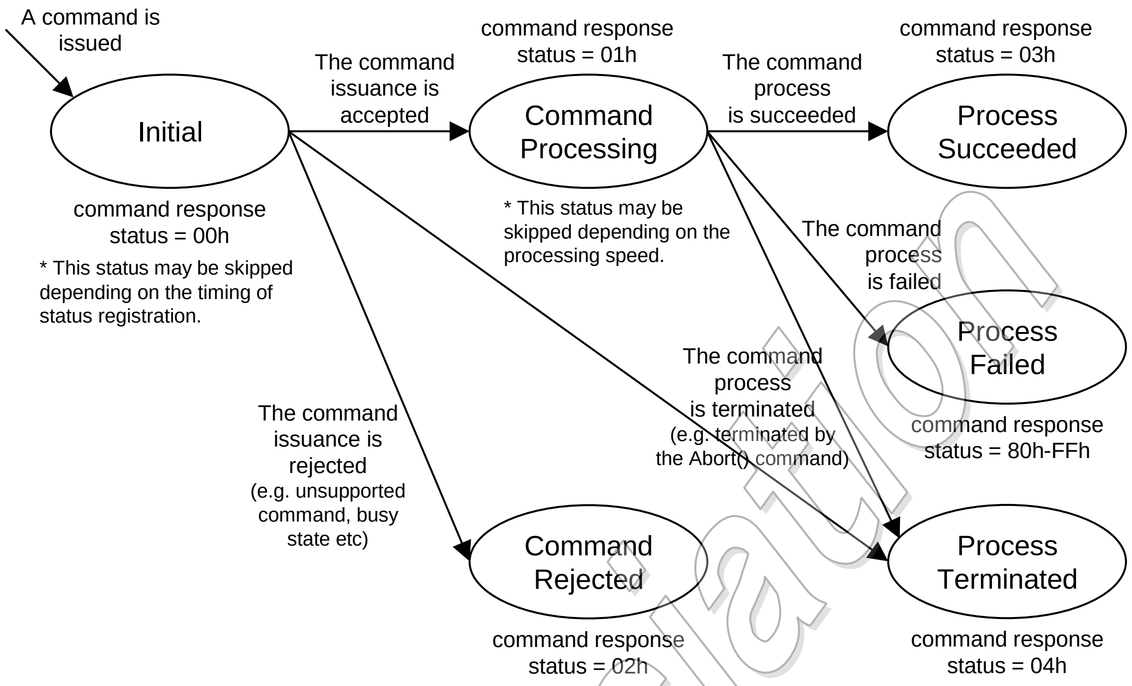
A command is issued
Initial
command response status = 00h
* This status may be skipped depending on the timing of status registration.
The command issuance is accepted
command response status = 01h
Command Processing
* This status may be skipped depending on the processing speed.
The command process is succeeded
command response status = 03h
Process Succeeded
The command process is failed
Process Failed
command response status = 80h-FFh
The command process is terminated (e.g. terminated by the Abort() command)
Process Terminated
command response status = 04h
The command issuance is rejected (e.g. unsupported command, busy state etc)
Command Rejected
command response status = 02h

Figure 2-2 : Command Response Status

2.3.3 Response Receiving

The following sequence defines the iSDIO response receiving.

(1) If a Host is able to confirm the command processing has succeeded, a Host receives response data corresponding to the issued command by reading iSDIO Command Response Data (**2.2.2.3 iSDIO Command Response Data**) via the Response Data Register Port.
By default the Response Data for the Command Response Status #1 in the Command Response Status Queue is available from the Response Register Port.

(2) If the "ReadResponse" command is issued, the Response Data for another Command Response Status (#2 to #8 and also #1) in the Command Response Status Queue becomes available from the Response Register Port.

(3) The Response Data Register Port is updated by the "ReadResponse" command or by an additional command if it is accepted. That is, the Response Data corresponding to the earlier command is overwritten by the Response Data corresponding to the later command.

(4) The iSDIO Command Response Data is read by CMD53 or CMD48. When CMD48 is used, if the iSDIO Command Response Data is equal to or smaller than 512 bytes, the data is read by one CMD48 command issuance (If the data is smaller than 512 bytes, unused bytes are filled with "00h") and if the data is larger than 512 bytes, it is necessary to be read in the unit of 512 bytes by multiple CMD48 command issuances. When CMD53 is used, the iSDIO Command Response Data is read by one CMD53 command issuance, however it is possible to be read by multiple CMD53 command issuances by dividing the data.

(5) Once the reading of the iSDIO Command Response Data is finished, no more data is available from the Response Register Port. At this time, if a Host reads from the Response Register Port, data of all "00h" is returned.

2.3.4 Buffer Overflow of Command Registration and Response Receiving

A Card has a buffer to store the Command Write Data from a Host and the buffer store the Response Data to a Host (for example, the iSDIO Wireless LAN Card receives the HTTP response message from a server to return the Command Response Data). However, the size of the buffer is limited and different between Cards, so a Host should determine the size of the buffer by reading the “Max Size of Command Write Data” or “Max Size of Command Response Data” in the iSDIO Capability Register. See **2.2.2.4 iSDIO Capability Register**.

When a Host writes the Command Write Data, if the size of the writing data exceeds the buffer size (“Max Size of Command Write Data”), the command registration shall fail and “CWE (Command Write Error)” in the Error Status is set to ‘1b’. A Host should reset “CWE” to ‘0b’ to write the next Command Write Data.

For a Card to receive the data, if the size of the receiving data exceeds the buffer size (“ Max Size of Command Response Data”), the Card stops the receiving and stores the received data in the buffer though all data is not prepared, and then the Command Response Status becomes in the “Process Succeeded” state and “CRU (Command Response Update)” in the iSDIO Status is set to ‘1b’, but “RRE (Response Receive Error)” in the Error Status is set to ‘1b’. (“ESU (Error Status Update)” and “CRE (Command Response Error)” in the Status Register are not changed)

3. Embedded iSDIO

An Embedded iSDIO Device is a product that utilizes the SD bus and SD commands.
"Embedded" is defined as a permanently soldered on a PCB (Printed Circuit Board) non-removable device or mounted through a socket permanently soldered on a PCB such that the Embedded iSDIO Device can't be removed by end users.

The Embedded iSDIO Device shall comply with **17. Embedded SDIO** in [SDIO] with the additional definitions.

- The SPI mode is not supported in the Embedded iSDIO Devices.
- The CIS is optional for the Embedded iSDIO Devices which supports the SDIO Command Access by using CMD52 and CMD53. The CIS is not required for the Embedded iSDIO Devices which supports the SD Memory Command Access by using CMD48 and CMD49.

4. Adapter-Type iSDIO Card

The iSDIO Card has a memory device and one or more I/O devices. Here, the memory device may be an internal NAND Memory module or an external (micro) SD Memory Card.

In the case of using the external (micro) SD Memory Card, the iSDIO Card has a card slot for the external (micro) SD Memory Card and is regarded as an adapter type of an iSDIO Card. If this adapter-type iSDIO Card without the external (micro) SD Memory Card is inserted into a Host, the iSDIO Card shall not provide iSDIO functions because the iSDIO functions need a memory device. In other words, the adapter-type iSDIO Card without the external (micro) SD Memory Card shall behave as follows.

- The iSDIO Card shall not indicate ready as the response of ACMD41. (The Card shall not complete memory initialization)
- The iSDIO Card shall complete I/O initialization, but shall not set IORx (I/O Ready) to '1b' for iSDIO function. (The Card shall not allow to use the iSDIO function)

Appendix A (Normative) : Reference

A.1 Reference

This specification refers the following documents.
Note that this specification requires at least the functions defined in the normative references to realize the application and functions defined in this specification.

[SDPart1] SD Specifications Part 1 Physical Layer Specification Version 4.10 or later

[SDPart2] SD Specifications Part 2 File System Specification Version 3.00 or later

[SDIO] SD Specifications Part E1 SDIO Specification Version 4.00 or later

Appendix B (Normative) : Special Terms

B.1 Terminology

Card	An SD card compliant with iSDIO specifications
Host	An SD host device compliant with iSDIO specifications
[iSDIO] Command	A command issued from a Host to a Card via the iSDIO Command Write Register Port
[iSDIO] Status	Status information read from a Card for a Host via the iSDIO Status Register
[iSDIO] Command Write Data	The data format for a Host to issue an iSDIO Command to a Card
[iSDIO] Command Response Data	The data format for a Host to read the Response Data corresponding to the issued iSDIO Command from a Card
[iSDIO] Status Register	The register for a Host to read the iSDIO Status from a Card
Command Write Register Port	The data port for a Host to write the iSDIO Command Write Data
Response Data Register Port	The data port for a Host to read from the iSDIO Command Response Data
Capability Register	The register for a Host to read the capability of a Card
CMD48	A Function Extension command to read the data in the iSDIO Register for the Card, which is defined in [SDPart1]
CMD49	A Function Extension command to write the data in the iSDIO Register for the Card, which is defined in [SDPart1]
CMD52	An SDIO command to read and write the one-byte data in the iSDIO Register for the Card, which is defined in [SDIO]
CMD53	An SDIO command to read and write the multi-byte data in the iSDIO Register for the Card, which is defined in [SDIO]
Embedded iSDIO	I/O devices that are embedded in a Host that utilizes the iSDIO electrical and command interface, but are not intended to be removed.