SD Specifications

Part E2

SDIO Card Bluetooth Type-A

Simplified Specification

Version 1.00
April 3, 2006

Technical Committee
SD Card Association

Revision History

Date	Version	Explanation and changes compared to previous issue
April 3, 2006	1.00	Simplified version initial release

Release of SD Simplified Specification

The following conditions apply to the release of the SD simplified specification ("Simplified Specification") by the SD Card Association. The Simplified Specification is a subset of the complete SD Specification which is owned by the SD Card Association.

Publisher:

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

Copyright Holder:

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

Disclaimers:

Conventions Used in This Document

Naming Conventions

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

1. Introduction....1
1.1 Background of the SDIO Card Specification for Bluetooth 1
1.2 Typical Applications....1
1.3 Type-A Card and Type-B Card 3
1.4 Requirements for Type-A Specification 4
2. Type-A Specification Overview 5
2.1 Scope 5
2.2 SDIO Card Type-A for Bluetooth Interface Structure 5
2.3 Supported Bluetooth Protocol Layers in the SDIO Card Type-A for Bluetooth....8
3. Card Management Interface 9
3.1 Overview 9
3.2 I/O Function Capability....9
3.3 Transfer Mode Capability 9
4. Type-A Transport 10
4.1 Overview 10
4.2 Packet Format....10
4.3 Adaptation to SDIO....11
4.3.1 Register Map....11
4.3.2 Byte Basis Transfer Mode....13
4.3.3 Block Basis Transfer Mode 15
5. CIS Formats....17
5.1 CIS Overview 17
5.2 CISTPL_SDIO_FUNCTION: SDIO Card for Bluetooth Supported Sub-Tuple....17
5.2.1 TPL_SDIOBT_RTC: SDIO Card for Bluetooth Retry Control Support 17
A.1 Normative References 18
B.1 Abbreviations and Terms 19
C.1 Example of SDIO Card Type-A for Bluetooth Initialization Procedure 21
D.1 Example Sequence of SDIO Adaptation Layer Operation....22
5.3 Byte Basis Transfer Mode....22
5.3.1 Write Flow – No Error (Byte Basis Transfer Mode)....22
5.3.2 Write Flow – CRC Error (Byte Basis Transfer Mode) 23
5.3.3 Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – No Error (Byte Basis Transfer Mode) 24
5.3.4 Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – No Error (Byte Basis Transfer Mode)....25
5.3.5 Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – CRC Error (Byte Basis Transfer Mode)....26
5.3.6 Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – CRC Error (Byte Basis Transfer Mode)....27
5.4 Block Basis Transfer Mode 28
5.4.1 Write Flow – No Error (Block Basis Transfer Mode) 28
5.4.2 Write Flow – CRC Error (Block Basis Transfer Mode)....29
5.4.3 Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – No Error (Block Basis Transfer Mode) 30
5.4.4 Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – No Error (Block Basis Transfer Mode)....31
5.4.5 Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – CRC Error (Block Basis Transfer Mode)....32
5.4.6 Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – CRC Error (Block Basis Transfer Mode)....33

Table of Figures

Figure 1 – Typical Applications in Data Communications 1
Figure 2 – Typical Applications in Audio/Visual Streaming.. 2
Figure 3 – Conceptual View of Type-A Function and Type-B Function.. 3
Figure 4 – Functional Relations between the Type-A Card and the Type-B Card....................................... 4
Figure 5 – SDIO Card Type-A for Bluetooth Interface Structure 6
Figure 6 – Two-Layered Aspect of the Type-A Interface Structure.. 7
Figure 7 – Typical Applications in Data Communications 8
Figure 8 – Type-A Transport Packet Write Operation from the Host to the Card in Byte Basis Transfer Mode 13
Figure 9 – Type-A Transport Packet Read Operation from the Card to the Host in Byte Basis Transfer Mode 14
Figure 10 – Type-A Transport Packet Write Operation from the Host to the Card in Block Basis Transfer Mode 15
Figure 11 – Type-A Transport Packet Read Operation from the Card to the Host in Block Basis Transfer Mode 16
Figure 12 – Initialization Procedure 21
Figure 13 – Write Flow – No Error (Byte Basis Transfer Mode).. 22
Figure 14 – Write Flow – CRC Error (Byte Basis Transfer Mode) 23
Figure 15 – Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – No Error (Byte Basis Transfer Mode) 24
Figure 16 – Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – No Error (Byte Basis Transfer Mode) 25
Figure 17 – Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – CRC Error (Byte Basis Transfer Mode) 26
Figure 18 – Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – CRC Error (Byte Basis Transfer Mode) 27
Figure 19 – Write Flow – No Error (Block Basis Transfer Mode) 28
Figure 20 – Write Flow – CRC Error (Block Basis Transfer Mode).. 29
Figure 21 – Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – No Error (Block Basis Transfer Mode) 30
Figure 22 – Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – No Error (Block Basis Transfer Mode) 31
Figure 23 – Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – CRC Error (Block Basis Transfer Mode) 32
Figure 24 – Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – CRC Error (Block Basis Transfer Mode) 33

Table of Tables

Table 1 – Type-A Transport Packet Format.. 10
Table 2 – Description of Service ID Field.. 10
Table 3 – SDIO Card Type-A for Bluetooth Register Map...11
Table 4 – SDIO Card Type-A for Bluetooth Register Description...11
Table 5 – CISTPL_SDIO_FUNCTION Tuple Definition 17
Table 6 – TPL_SDIOBT_RTC Field Values 17

1. Introduction

1.1 Background of the SDIO Card Specification for Bluetooth

The SDIO (Secure Digital I/O) Card is based on and compatible with the SD Memory Card and it extends SD applications from memory to I/O. Bluetooth is a short-range wireless communication protocol which has features such as compact, low power consumption and low cost. Both SDIO and Bluetooth are designed to be compact and slim for portable appliances. An SDIO Card for Bluetooth is one of the primary applications for SDIO.

1.2 Typical Applications

An SDIO Card for Bluetooth can be used in a wide range of appliances, ranging from cellular phones, PDAs and notebook computers to AV equipment such as digital TVs and audio players.

Figure 1 provides typical applications in data communications. Most users access the Internet via multiple network devices. They utilize the wired network in office and they use the cellular network when in outdoor areas. The swappable feature of SDIO is very suitable for such usages.

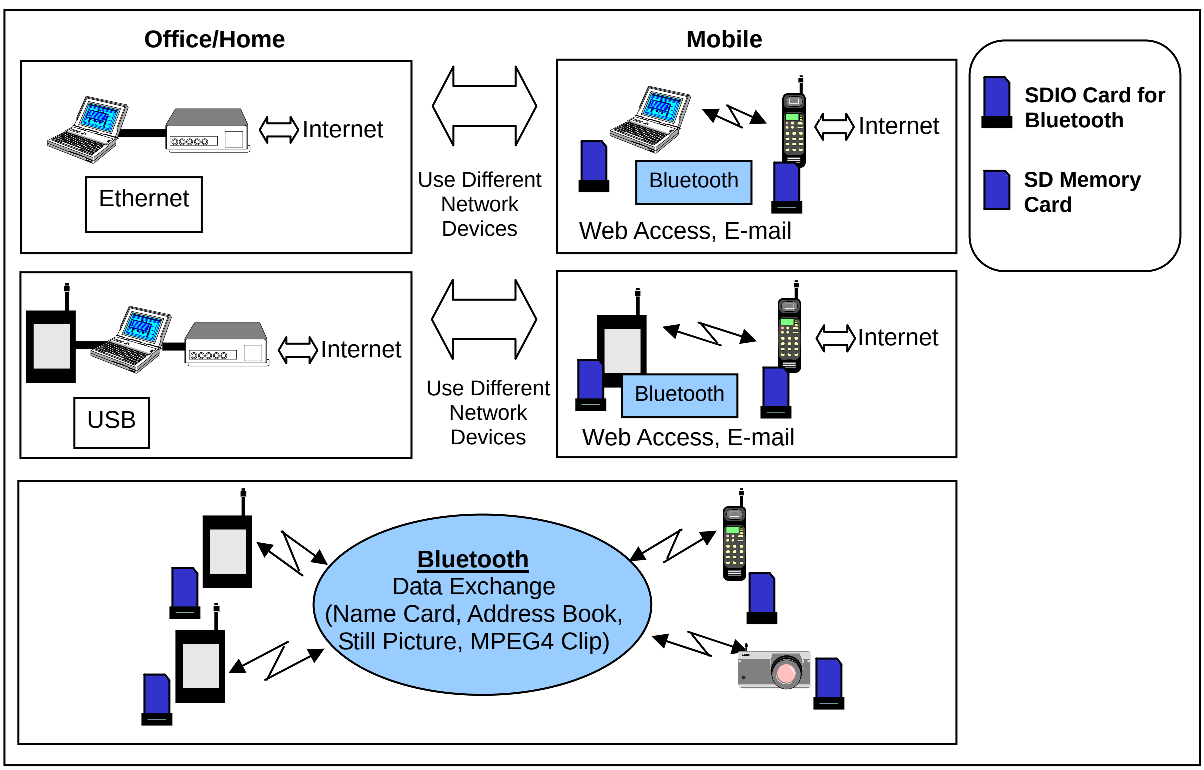

Figure 1 – Typical Applications in Data Communications

Figure 2 provides typical applications in AV streaming. While currently most users cannot afford to pay expensive charges to listen to music or watch video over the cellular network, they can utilize an SD Memory Card to do so. When they are at home, they can use an SDIO Card for Bluetooth to listen to music and watch video from home-based multimedia servers. This swappable feature between SD Memory Cards and SDIO Cards for Bluetooth are very suitable for such usage.

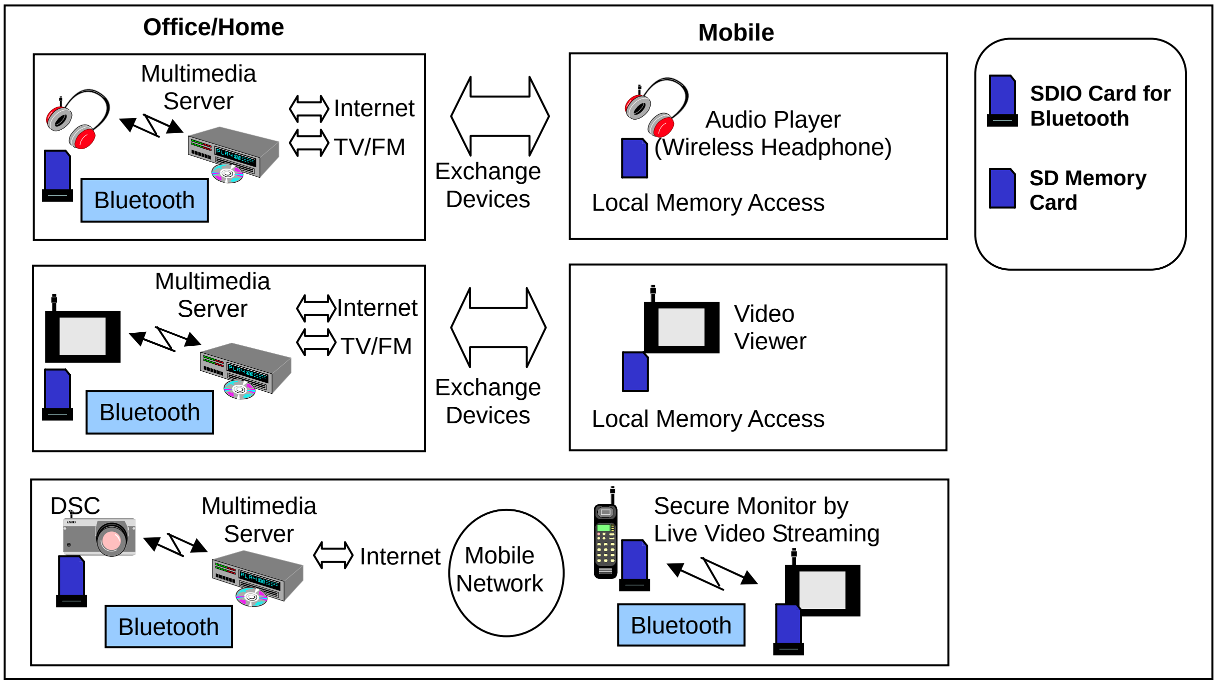

Figure 2 – Typical Applications in Audio/Visual Streaming

1.3 Type-A Card and Type-B Card

An SDIO Card for Bluetooth provides Bluetooth wireless technology in the form of SD Card dimensions. The SDIO Card Type-A Specification for Bluetooth provides a uniform interface for accessing an SDIO Card Type-A for Bluetooth. This interface enables the Host and the Card to execute Bluetooth protocols cooperatively in such a way that some layers reside in the Host and the rest in the Card. It defines the service provided by the Bluetooth protocol layers that reside in the Card, and the data transport mechanism over SDIO.

These slim cards come with two different implementations to cater to various kinds of appliances. The Type-A implementation is meant for appliances like PDAs or notebook computers that have sufficient processing resources and are easily re-configurable. The Type-B implementation is intended for appliances that have very limited processing resources or are difficult to re-configure, such as audio players, video viewers and Digital Still Cameras (DSC).

Figure 3 shows a conceptual view of the Type-A function and the Type-B function. The Type-A implementation consists of only the Radio, Baseband and the SDIO interface. The remaining Bluetooth protocol layers and profiles reside in the Host. The Type-B implementation consists of those modules found in Type-A implementation plus additional upper Bluetooth protocol layers such as L2CAP, SDP and RFCOMM.

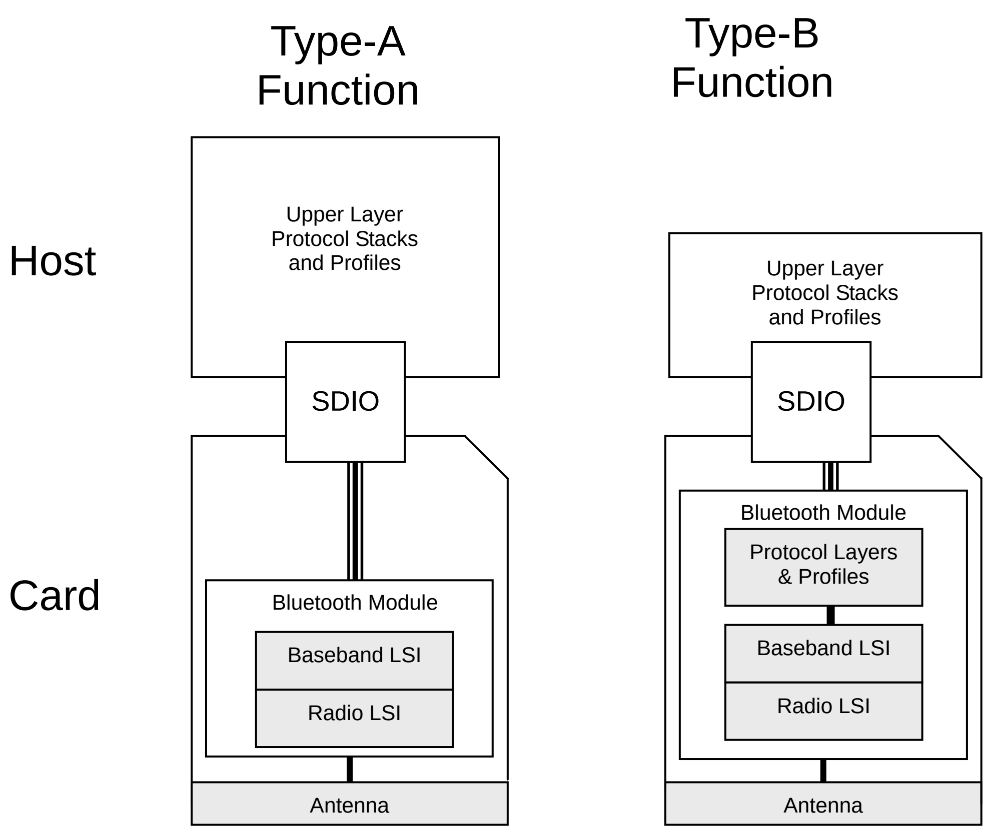

Figure 3 – Conceptual View of Type-A Function and Type-B Function

Figure 4 shows the relationship between the Type-A Card and the Type-B Card. The Type-B Card supports both Type-A and Type-B functions, while the Type-A card provides only the lowest layers of the Bluetooth stack. The Type-B Card can execute either the Type-B function or the Type-A function at any time depending on the requirements of the Host. The Type-A Card does not contain the Type-B function capabilities.

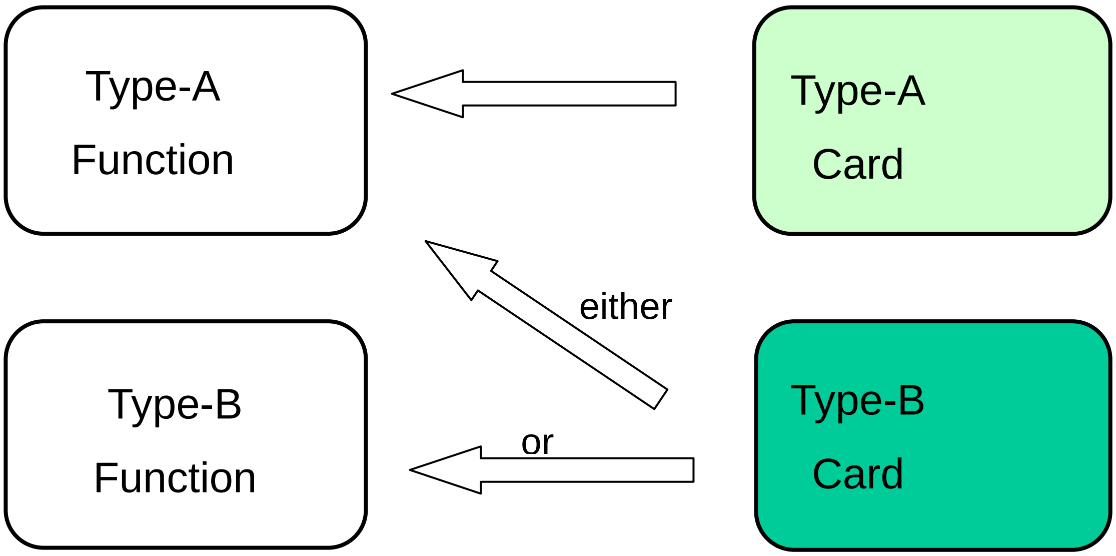

Figure 4 – Functional Relations between the Type-A Card and the Type-B Card

1.4 Requirements for Type-A Specification

The SDIO Card Type-A Specification for Bluetooth is intended for devices that have sufficient processing resources and can easily be updated or reconfigured. To support these devices, the SDIO Card Type-A Specification for Bluetooth shall have the following requirements:

- Minimize design change in existing host software and existing host hardware
- Decrease Bluetooth specific design for host developers
- Allow use of any Baseband and radio

2. Type-A Specification Overview

2.1 Scope

This document describes the functional specification for the interface between a Host and an SDIO Card for Bluetooth over the SDIO (Secure Digital I/O) transport. That is, it defines service interfaces provided by the Bluetooth protocol in the card side, and the transport mechanism for service interface data over SDIO. Physical, Electrical and Mechanical aspects of the interface are defined in the SDIO specification and this document refers to '*SDIO Card Specification*' for relevant information.

This document covers the specification for the SDIO Card Type-A for Bluetooth. For the specification of SDIO Card Type-B for Bluetooth, refer to the document '*SDIO Card Type-B Specification for Bluetooth*'.

The Bluetooth protocol specification is based on the document '*Specification of the Bluetooth System Version 1.1*' by the Bluetooth Special Interest Group (SIG).

2.2 SDIO Card Type-A for Bluetooth Interface Structure

Figure 5 shows the SDIO Card Type-A for Bluetooth interface structure. The Type-A card provides only an HCI interface to the Host.

The Type-A interface provides the Host with a transport for the Bluetooth HCI Protocol. It uses SDIO commands to provide a reliable and an unreliable transfer mode capability.

The Bluetooth Protocol Service Interface has a two-layer architecture. The bottom layer is SDIO and the top layer is the SDIO Type-A for Bluetooth Transport. The Type-A Transport provides a packet-oriented transport interface between the upper protocol layers in the Host and lower protocol layers in the Card. The packet-oriented service keeps packet boundaries in transmitting and receiving data over SDIO. It provides a protocol-multiplexing capability and can also offer an error free transport via two SDIO transfer modes (Byte Basis or Block Basis). Figure 6 shows the two-layered aspect of the Type-A interface.

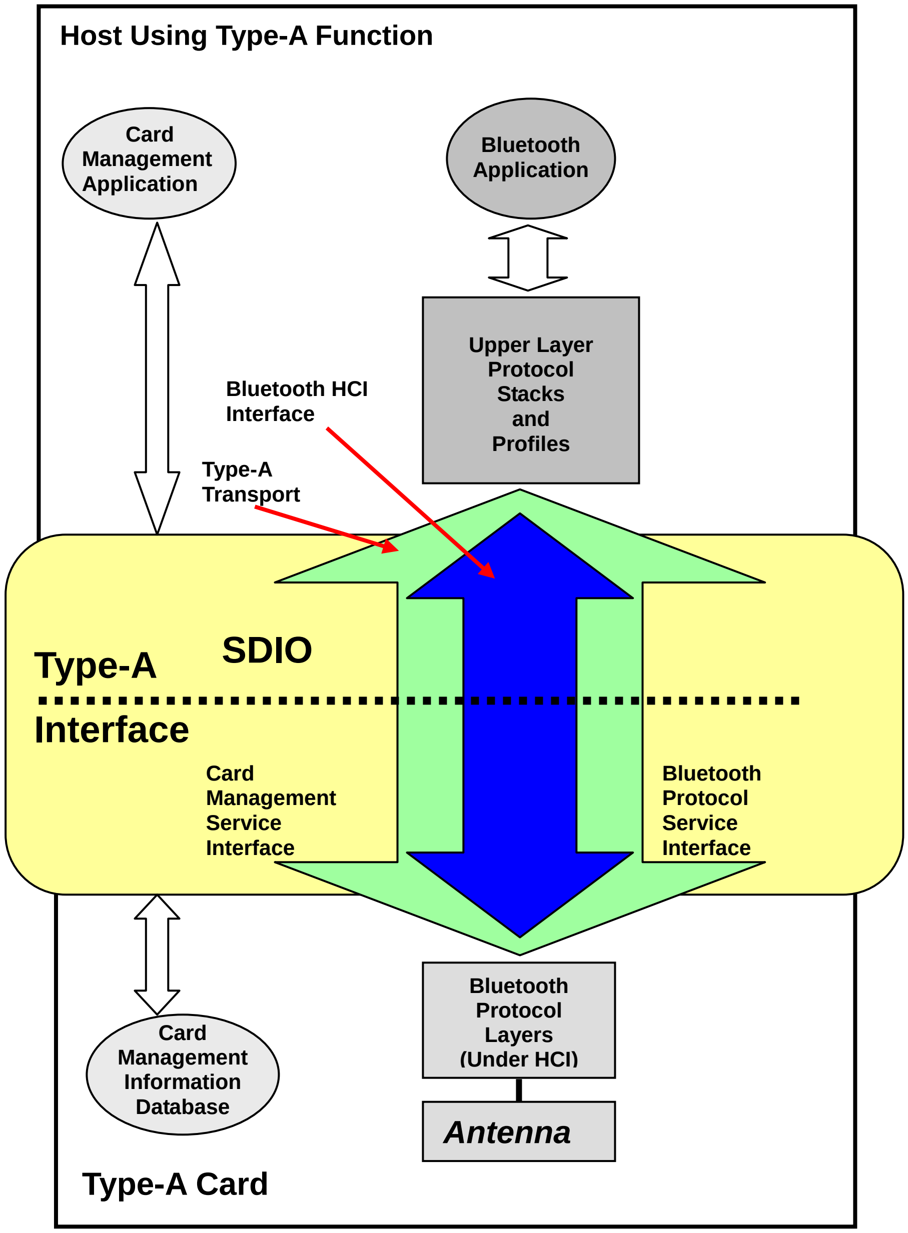

Figure 5 – SDIO Card Type-A for Bluetooth Interface Structure

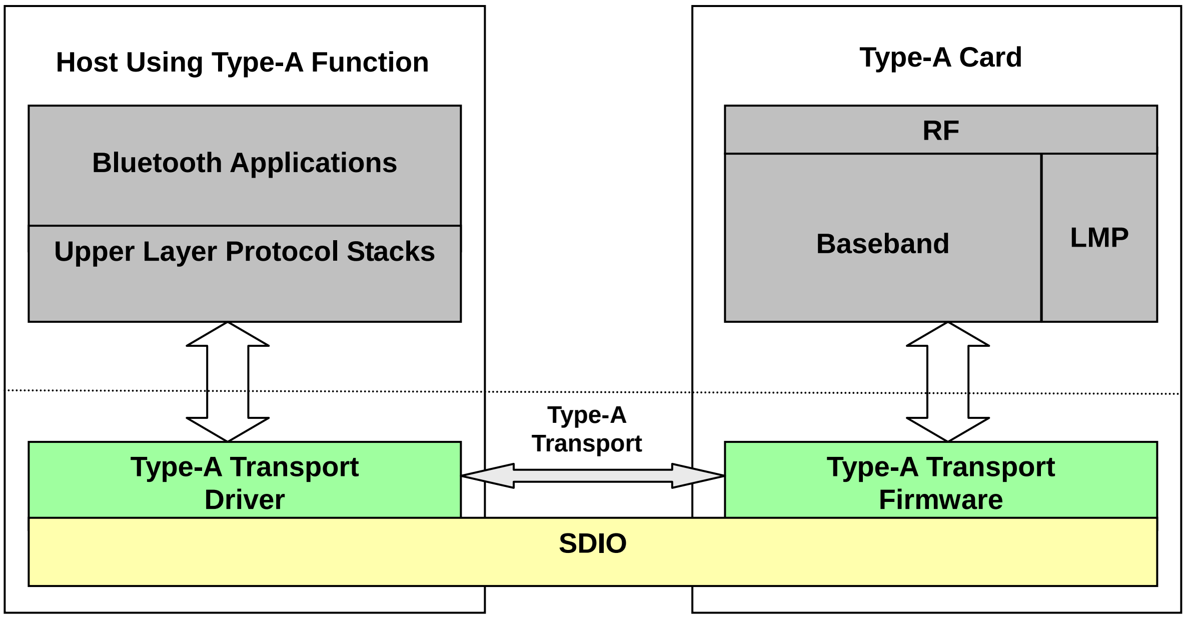

Figure 6 – Two-Layered Aspect of the Type-A Interface Structure

2.3 Supported Bluetooth Protocol Layers in the SDIO Card Type-A for Bluetooth

Figure 7 shows the supported Bluetooth protocol layer in the SDIO Card Type-A for Bluetooth. The radio, Baseband, LMP and HCI interface reside in the card. The Bluetooth protocol specification is based on the document '*Specification of the Bluetooth System Version 1.1*' by the Bluetooth Special Interest Group (SIG).

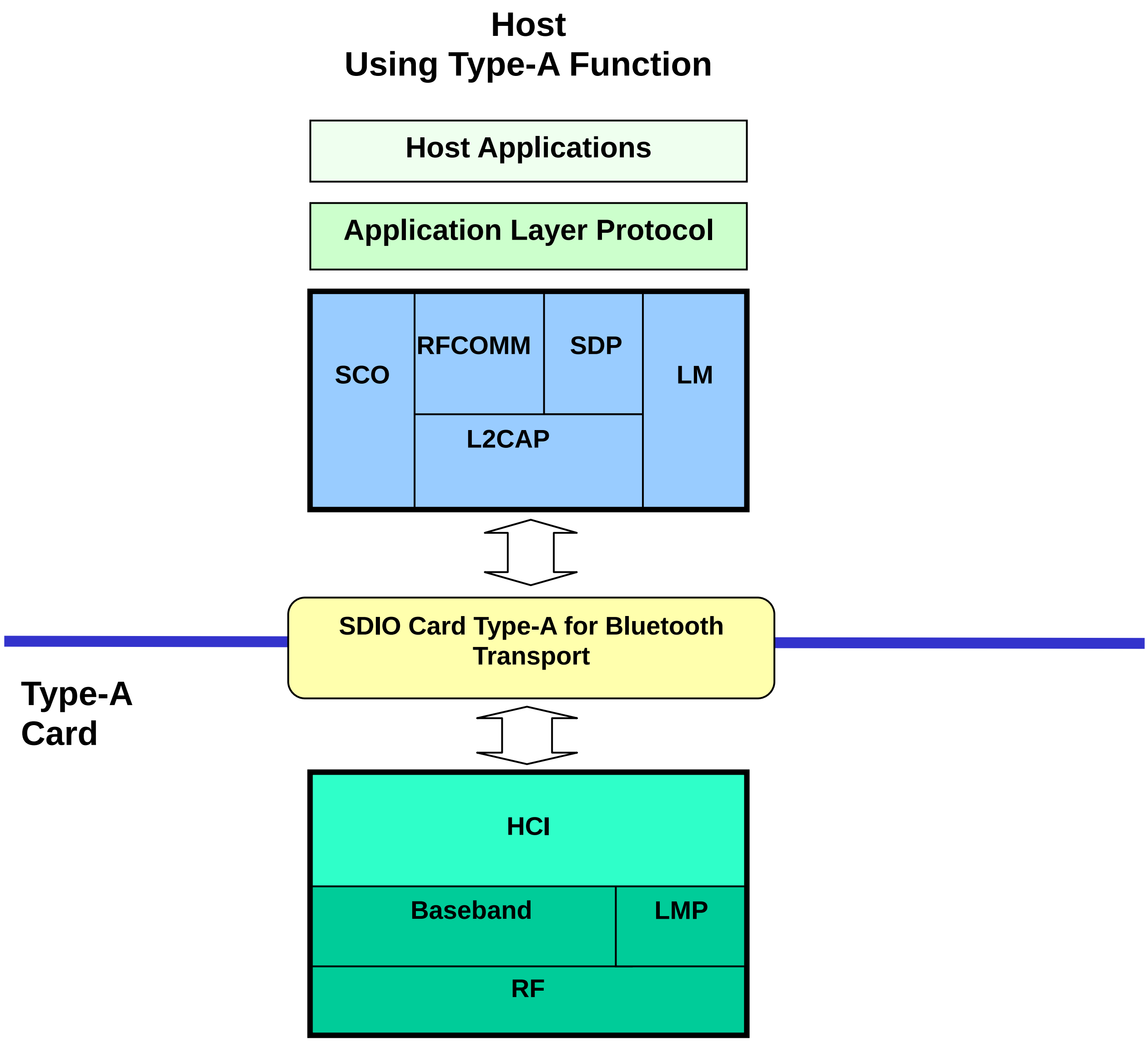

Figure 7 – Typical Applications in Data Communications

3. Card Management Interface

3.1 Overview

The Card Management Service provides the Host with card I/O function capability (Type-A or Type-B) and SDIO transfer mode capability. This information can be accessed via various defined SDIO registers.

3.2 I/O Function Capability

The I/O device interface field in the Function Basic Register (FBR) provides information on the I/O function capability. The Host can recognize whether an inserted card is a Type-A Card or a Type-B Card. The CIS (Card Information Structure) area provides configuration information for the Type-A card. Please refer to the document '*SDIO Card Specification*' for both FBR and CIS information. The Host normally accesses these registers in early stages of card initialization procedure and it makes sure that the Host can communicate with the Card. The Host will then proceed to initialize the Type-A Transport. Refer to Appendix C for an example of the initialization procedure.

3.3 Transfer Mode Capability

The SMB (Supports MBIO) field in the Common Card Control Register (CCCR) provides Block Basis transfer capability. The Host can recognize whether the inserted card is capable of Block Basis transfer (optional). Please refer to '*SDIO Card Specification*' for CCCR information.

4. Type-A Transport

4.1 Overview

The Type-A Transport provides a packet-oriented transport on SDIO. That is, the Type-A Transport packet boundary is kept, both when writing data to the Card and reading data from the Card. It also provides the Type-A Host Control Interface multiplexing functions and adaptation functions to offer error-free transportation over two SDIO transfer modes - Byte Basis and Block Basis.

4.2 Packet Format

Table 1 shows the Type-A Transport packet format. Packet Length is used to identify the length of the Interface Data field in the packet and includes the Packet Length and Service ID fields. Service ID shows the protocol layer that issues the interface data, that is, multiple Bluetooth HCI protocol services are multiplexed in this service interface. The maximum length of a Type-A Transport packet is 65543 (65535 + 4 + 4) bytes, the maximum defined in the *Specification of the Bluetooth System Version 1.*1 for an HCI packet, plus the packet header. The packets are sent in LSB (Least Significant Byte) first and MSB (Most Significant Byte) last manner, commonly called little endian.

Table 1 – Type-A Transport Packet Format

Packet Length	Service ID	Interface Data
3 Bytes	1 Byte	N Bytes

Table 2 shows the description of the Service ID field. Contents of the Interface Data depend on the Service ID.

Table 2 – Description of Service ID Field

Service ID	Description
0x00	Reserved
0x01	Data is Bluetooth HCI Command Packet
0x02	Data is ACL Data Packet
0x03	Data is SCO Data Packet
0x04	Data is Bluetooth HCI Event Packet
0x05-0xFD	Reserved
0xFE	Vendor Unique
0xFF	Reserved

4.3 Adaptation to SDIO

4.3.1 Register Map

Table 3 shows the SDIO Card Type-A register map and Table 4 shows their descriptions. There is one register for selecting card operational mode (not used in Type-A mode); three registers for interrupt control and four registers for Read/Write control (counting Receiver Data and Transmitter Data as separate registers).

Table 3 – SDIO Card Type-A for Bluetooth Register Map

Add	RW	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00	RO	Receiver Data	RDAT							
	WO	Transmitter Data	TDAT							
0x10	WO	Read Packet Control	0	0	0	0	0	0	0	PC RRT
0x11	WO	Write Packet Control	0	0	0	0	0	0	0	PC WRT
0x12	RO	Retry Control Status	0	0	0	0	0	0	0	RTC STAT
	WO	Retry Control Set	0	0	0	0	0	0	0	RTC SET
0x13	RO	Interrupt Identification	0	0	0	0	0	0	0	INTRD
	WO	Interrupt Clear	0	0	0	0	0	0	0	CL INTRD
0x14	RW	Interrupt Enable	0	0	0	0	0	0	0	EN INTRD
0x20	RO	Bluetooth Mode Status	0	0	0	0	0	0	0	MD STAT

Table 4 – SDIO Card Type-A for Bluetooth Register Description

Field	Type	Description
RDAT	RO	Access Window to Receive Data: Any read from this address will pass data from the byte addressed by the read address pointer. After any read from this register, the read address pointer will be automatically incremented by 1. The Host shall not use CMD52 and shall use CMD53 for accessing RDAT (Address 0x00). When the Host uses CMD52 for accessing RDAT, the Card will return 'OUT_OF_RANGE' error to the Host in SD mode or 'PARAMETER ERROR' in SPI mode.
TDAT	WO	Access Window to Transmit Data: Any write to this address will pass data to the byte addressed by the write address pointer. After any write to this register, the write address pointer will be automatically incremented by 1. The Host shall not use CMD52 and shall use CMD53 for accessing TDAT (Address 0x00). When the Host uses CMD52 for accessing TDAT, the Card will return 'OUT_OF_RANGE' error to the Host in SD mode or 'PARAMETER ERROR' in SPI mode.
PCRRT	WO	Packet Read Retry:

Field	Type	Description
		The PCRRT bit indicates that the Host requests the SDIO Card to send the current packet data again or the next packet data. This bit shall be set or reset after reading one packet data without exception. If this bit is set to 1, it indicates that the Host requests the current packet data again. The SDIO Card leaves the current packet data intact, it sets the read address pointer of the Packet Read Buffer back to the beginning of the current packet data, and it will send the interrupt to the Host. If this bit is reset to 0, it indicates that the Host requests the next packet data. The SDIO Card prepares for sending the next packet data, and then it will send the interrupt to the Host. This bit does not affect the Packet Write Buffer. This bit is auto cleared, so there is no need to rewrite a value of 0. This bit is write-only. Any read will return an undetermined value. Note: It is acceptable for the Host to set this bit even if there has been no error condition detected. In this case, the Card shall behave the same as if there had been an actual error detected. Additionally, the Host is not required to issue a retry when an error condition is detected. The Card shall be able to proceed to the next command.
PCWRT	WO	Packet Write Retry: The PCWRT bit indicates that the Host sends the current packet data to the SDIO Card again. If this bit is set to 1, the SDIO Card sets the write address pointer of the Packet Write Buffer back to the beginning of the current packet data. This bit does not affect the Packet Read Buffer. This bit is auto cleared, so there is no need to rewrite a value of 0. This bit is write-only. Any read will return an undetermined value. Note: It is acceptable for the Host to set this bit even if there has been no error condition indicated by the Card. In this case, the Card shall ignore the retry Type-A Transport packet. The Host shall issue a retry request upon detecting an error condition from the Card.
RTC STAT	RO	Retry Control Status: The Host reads this bit to determine that the Retry Control mode has been changed. The Host shall wait until this bit is read back as the same value that was written to the RTC SET bit, one or zero, before issuing another command to the Card. If the CIS does not indicate that the Card supports RTC, this bit will always be read as a zero.
RTC SET	WO	Retry Control Set: If the Card does not support Retry Control as indicated in the CIS, this bit is ignored. If the Card does support Retry Control, this bit is set to 1 by the host if it will NOT return a Read Acknowledgement to the Card. This bit can also be written with 0 to turn the Read Acknowledge back on. The Host should always check the RTC_STAT bit after writing this bit to verify that the mode change has been done.
INTRD	RO	Packet Read Ready Interrupt: This bit indicates that the Card has a packet ready to send to the Host when set to 1 and corresponds to the Card indicating an interrupt on the SDIO bus. This bit is used by the Host to identify the interrupt pending, and this bit will remain set to 1, until the Host writes 1 to the CLINTRD bit. This bit is set once for every packet. On power up or after a reset, this bit shall be set to 0.
CLINTRD	WO	Clear the INTRD Bit: If this bit is set to 1, the INTRD bit is cleared. The CLINTRD bit is auto cleared, so there is no need to rewrite a value of 0. This bit is write-only.
ENINTRD	RW	Enable the INTRD Bit: If this bit is cleared to 0, this indicates that the interrupt by the INTRD bit will not be sent to the Host. On Power up or after a reset, this bit shall be set to 0.
MDSTAT	RO	SDIO Card for Bluetooth Mode Status: Defines the SDIO Card for Bluetooth mode. On power up or after a reset, this bit shall be 0 to indicate Type-A card (HCI level only).

**Note*: Unused register Bit 1 to Bit 7 are set to 0.

4.3.2 Byte Basis Transfer Mode

This section describes the way of keeping packet boundary and recovering from CRC errors in Byte Basis transfer mode. See Appendix D for details of data transfer operation.

Figure 8 describes the Type-A Transport packet Write operation from the Host to the Card in Byte Basis transfer mode. When the Type-A Transport in the Host writes a packet to the Card through the SDIO Interface, it divides the Type-A Transport packet into several SDIO data packets. Assuming that L denotes the length of a Type-A Transport packet (in bytes) and B denotes the maximum byte count in Byte Basis transfer mode (B < L). There are [L / B] SDIO data packets to be sent. The Type-A Transport in the Host repeatedly sends all SDIO data packets using Byte Basis transfer mode until all the SDIO data packets are sent. In Figure 8, the maximum transfer size is used in each SDIO data. However, smaller byte counts may also be used for this purpose.

When a CRC error occurs in an SDIO transmission, the Type-A Transport in the Host recognizes it through the transmission result and it sets the PCWRT field to "1" in order to prepare for a re-write. Then it shall try to re-send the whole Type-A Transport data packet, including the erroneous SDIO data packet, until the erroneous packet is sent successfully or a number of retries designated by the Host are reached. If the packet is not sent successfully, the Type-A Transport sends a fatal error to the Host application. The Host application should reset the Type-A Transport to resume Bluetooth communication.

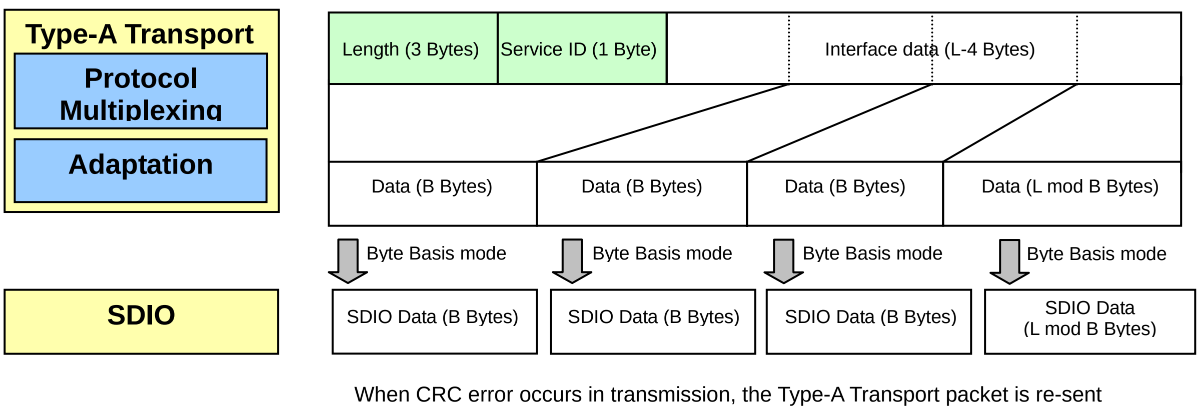

Figure 8 – Type-A Transport Packet Write Operation from the Host to the Card in Byte Basis Transfer Mode

Figure 9 describes the Type-A Transport packet Read operation from the Card to the Host in Byte Basis transfer mode. When the Type-A Transport in the Host receives an interrupt via the Packet Read Ready signal, it starts to read the Type-A Transport packet from the Card. This interrupt signal shall be issued for receipt of Type-A Transport packets instead of receipt of SDIO data packets. The Type-A Transport header is read first, and it is used to determine the length of the Type-A Transport packet. The Type-A Transport in the Host then repeatedly reads each SDIO data packet using Byte Basis transfer mode until the total length of the received SDIO data packets reaches the Type-A Transport packet length. It then sets the PCRRT to "0" to indicate the success of packet read operation to the Type-A Transport in the Card and assembles all SDIO data packets into one Type-A Transport packet. Finally it removes the Type-A Transport packet header and informs the Type-A Host Control Interface of arrival of data packet. In Figure 9, maximum transfer size is used in each SDIO data packets. However, smaller byte counts may also be used.

When a CRC error occurs in the SDIO transmission, the Type-A Transport in the Host recognizes it through the transmission result. It sets the PCRRT field to "1" in order to prepare for a re-read, and the Type-A Transport in the Card issues an interrupt to the Host. When the interrupt is cleared, it shall try to re-read the whole Type-A Transport data packet, including the erroneous SDIO data packet, until the erroneous packet is received successfully or a number of retries designated by the Host are reached. If the packet is not received successfully, the Type-A Transport sends a fatal error to the Host application. The Host application should reset the Type-A Transport to resume Bluetooth communication.

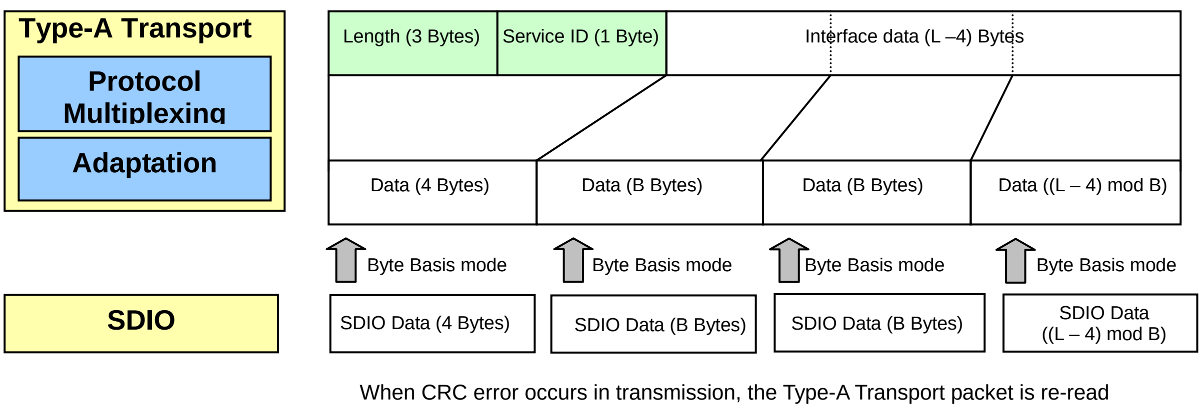

Figure 9 – Type-A Transport Packet Read Operation from the Card to the Host in Byte Basis Transfer Mode

4.3.3 Block Basis Transfer Mode

This section describes the way of keeping packet boundary and recovering from CRC errors in Block Basis transfer mode. Block Basis transfer mode has the advantages of less host resource consumption and high speed SDIO transfer due to elimination of host intervention during SDIO data transfer. Block Basis transfer mode is optional in the SDIO specification. However, the Type-A Transport requires an error free transport over SDIO, therefore error recovery procedure must be implemented if Block Basis transfer mode is supported.

Figure 10 describes the Type-A Transport packet Write operation from the Host to the Card in Block Basis transfer mode. When the Type-A Transport in the Host writes a packet to the Card through SDIO Interface, it divides the Type-A Transport packet into several SDIO data packets. Assuming that L denotes the length of Type-A transport packet (in bytes) and B denotes maximum byte count in Byte Basis transfer mode (B < L), there are [L / B] SDIO data packets to be sent. The Type-A Transport then writes the whole Type-A Transport packet as SDIO data packets concatenated with Block Basis transfer mode.

When a CRC error occurs in an SDIO transmission, the Type-A Transport in the Host determines it through the transmission result. It aborts multi bock transfer and sets the PCWRT field to "1" in order to prepare for a re-write. It shall then try to re-send the whole Type-A Transport data packet, until the erroneous packet is sent successfully or a number of retries designated by the Host are reached. If the packet is not sent successfully, the Type-A Transport sends a fatal error to the Host application. The Host application should reset the Type-A Transport to resume Bluetooth communication.

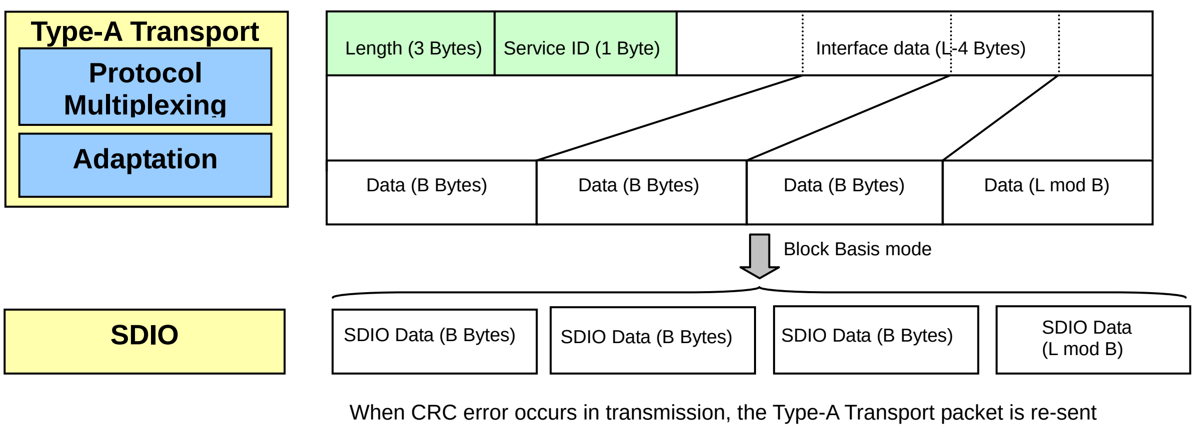

Figure 10 – Type-A Transport Packet Write Operation from the Host to the Card in Block Basis Transfer Mode

Figure 11 describes the Type-A Transport packet Read operation from the Card to the Host in Block

Basis transfer mode. When the Type-A Transport in the Host receives an interrupt from the Packet Read Ready signal, it starts to read the Type-A Transport packet from the Card. This interrupt signal shall be issued for receipt of Type-A Transport packets instead of receipt of SDIO data packets. The Type-A Transport header is read first, and it is used to determine the length of the Type-A Transport packet. The Type-A Transport in the Host then reads the entire Type-A Transport packet as SDIO data packets are concatenated in Byte Basis transfer mode until the total length of received SDIO data reaches the Type-A Transport packet length. It then sets the PCRRT to "0" to indicate the success of packet read operation to the Type-A Transport in the Card. Finally it removes the Type-A Transport packet header and informs the Type-A Host Control Interface of the arrival of a new data packet.

When a CRC error occurs in SDIO transmission, the Type-A Transport in the Host recognizes it through the transmission result. It aborts multi block transfer and sets the PCRRT field to "1" in order to prepare for a re-read, and the Type-A Transport in the Card issues an interrupt to the Host. When the interrupt is cleared, it shall try to re-read the Type-A Transport data packet, until the erroneous packet is received successfully or a number of retries designated by the Host are reached. If the packet is not received successfully, the Type-A Transport sends a fatal error to the Host application. The Host application should reset the Type-A Transport to resume Bluetooth communication.

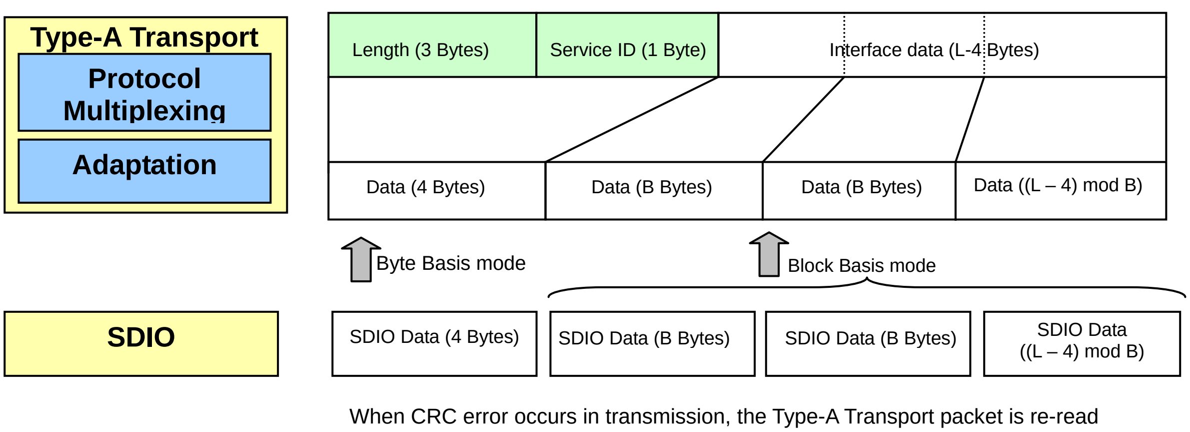

Figure 11 – Type-A Transport Packet Read Operation from the Card to the Host in Block Basis Transfer Mode

5. CIS Formats

5.1 CIS Overview

The SDIO Card for Bluetooth utilizes Sub-Tuples of the CISTPL_SDIO_FUNCTION Tuple as described in the SDIO specification. These Sub-Tuples are assigned a code of 0x91, which is in the 'Vendor Specific' area as defined by the PC Card specification. The TPL_SDIO_INTERFACE field is set to the FBR value (2) defined in the SDIO Card Type-A Specification for Bluetooth. There is one data value used in this Tuple to inform the Host if the Retry Control Bit is supported on this card.

5.2 CISTPL_SDIO_FUNCTION: SDIO Card for Bluetooth Supported Sub-Tuple

The CISTPL_SDIO_FUNCTION Tuple is used to inform the host if the card supports the Retry Control Bit and functionality. This Tuple is optional; if it is not present the Card requires the Read Acknowledgement to be sent.

Table 5 – CISTPL_SDIO_FUNCTION Tuple Definition

Byte	Size	Description
0x00	1	TPL_CODE_CISTPL_RTC_SUPPORTED (0x91)
0x01	1	TPL_LINK Link to the next Tuple (0x03)
0x02	1	TPL_SDIO_INTERFACE (0x02 for Type-A Bluetooth)
0x03	1	TPL_STANDARD (0x00)
0x04	1	TPL_SDIOBT_RTC (0x01 or 0x00)

5.2.1 TPL_SDIOBT_RTC: SDIO Card for Bluetooth Retry Control Support

The field denotes whether or not the Card does not require a Read Acknowledge after the Host has done a read block.

Table 6 – TPL_SDIOBT_RTC Field Values

Value	Description
0x00	The Card requires the Read Acknowledge to be sent
0x01	The Card does not require the Read Acknowledge to be sent
0x02-0xFF	Reserved

Appendix A (Normative)

A.1 Normative References

The following documents are referenced by this specification. The reader is directed to the respective owners to obtain copies.

1) SD Memory Card Specification
Part 1 PHYSICAL LAYER SPECIFICATION
Version 1.01, April 2001

2) SD Card Specification
Part E1 Secure Digital Input/Output (SDIO) Card Specification
Version 1.00, October 2001

3) Specification of the Bluetooth System
Version 1.1, February 22, 2001
Bluetooth Special Interest Group (SIG)

4) SDIO Card Type-B Specification for Bluetooth
Version 1.00 Draft E, September 2002

Appendix B (Normative)

B.1 Abbreviations and Terms

AV	Audio/Visual
ACL	Asynchronous Connection-Less
BD_ADDR	Bluetooth Device Address
Block	A Number of Bytes, Basic Data Transfer Unit
CCCR	Common Card Control Register
CIA	Common Information Area
CIS	Card Information Structure
CLINTRD	Clear the INTRD Bit
CMD	Command Line or SD Bus Command (if extended CMDXX)
CRC	Cyclic Redundancy Check
CSA	Code Storage Area
DAT	Data
DSC	Digital Still Camera
ENINTRD	Enable the Interrupt Bit
FBR	Function Basic Registers
HCI	Host Controller Interface
ID	Identity
INTRD	Packet Read Ready Interrupt
I/O	Input/Output
L2CAP	Logical Link Control and Adaptation Protocol
LAP	Lower Address Part
LM	Link Manager
LMP	Link Manager Protocol
LSB	Least Significant Byte
MDSTAT	SDIO Card for Bluetooth Mode Status

MSB	Most Significant Byte
PCRRT	Packet Read Retry
PCWRT	Packet Write Retry
R/O	Read Only
R/W	Read or Write
RDAT	Receive Data
RF	Radio Frequency
RFCOMM	Serial Cable Emulation Protocol based on ETSI TS 07.10
RTC	Retry Control
RTC STAT	Retry Control Status
RTC SET	Retry Control Set
SCO	Synchronous Connection-Oriented
SD	Secure Digital
SDA	SD Association
SDIO	Secure Digital I/O
SIG	Bluetooth Special Interest Group
SMB	Supports MBIO
SPI	Serial Peripheral Interface
STAT	Status
TDAT	Transmit Data
Tuple	Data Blocks in a Linked List or Chain Format
W/O	Write Only

Appendix C (Informative)

C.1 Example of SDIO Card Type-A for Bluetooth Initialization Procedure

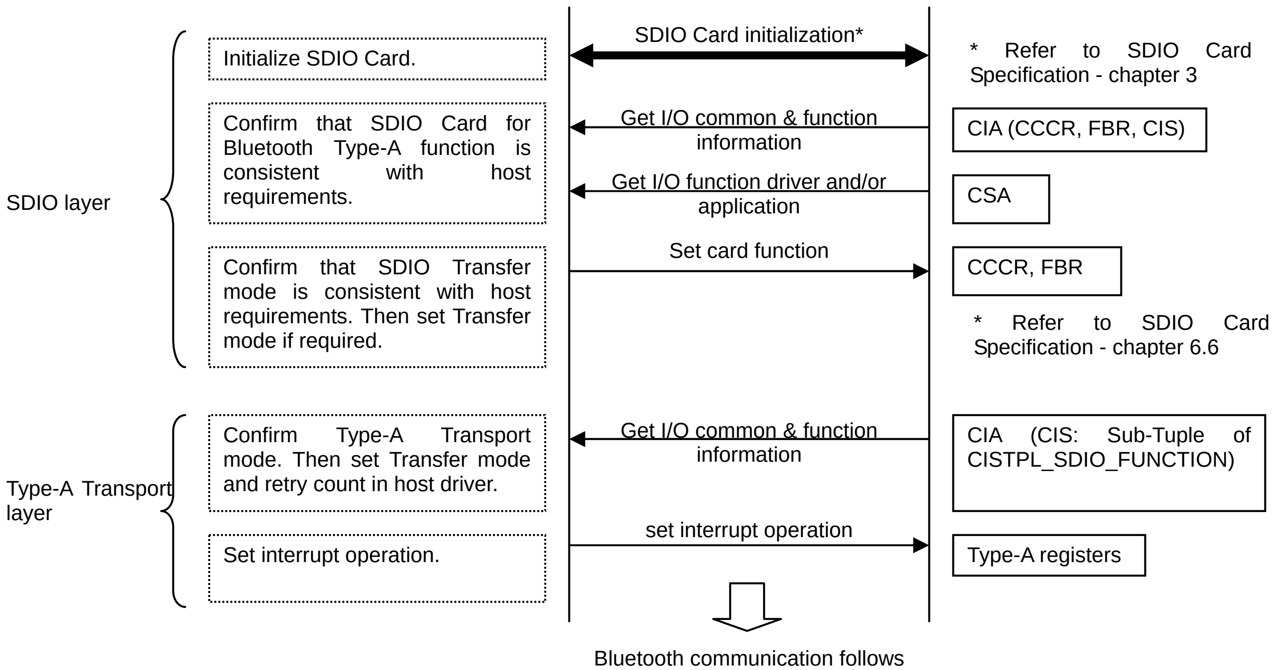

Figure 12 – Initialization Procedure

Appendix D (Informative)

D.1 Example Sequence of SDIO Adaptation Layer Operation

5.3 Byte Basis Transfer Mode

5.3.1 Write Flow – No Error (Byte Basis Transfer Mode)

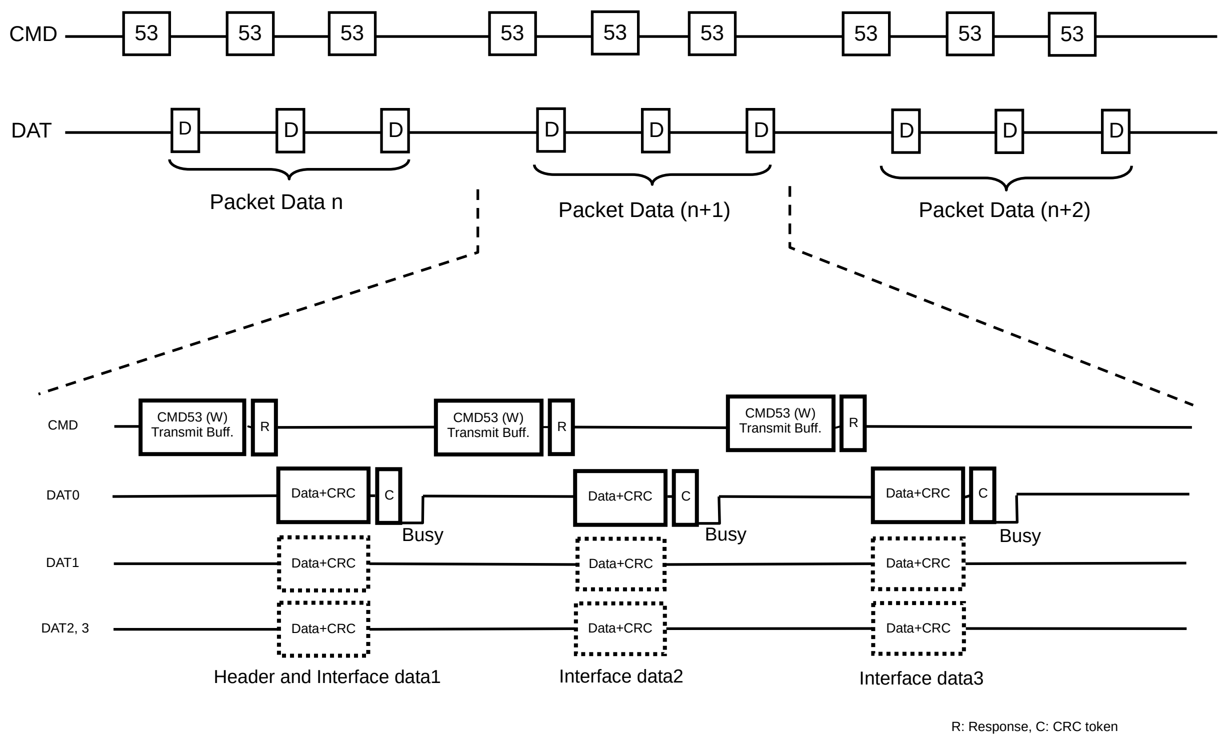

Figure 13 – Write Flow – No Error (Byte Basis Transfer Mode)

5.3.2 Write Flow – CRC Error (Byte Basis Transfer Mode)

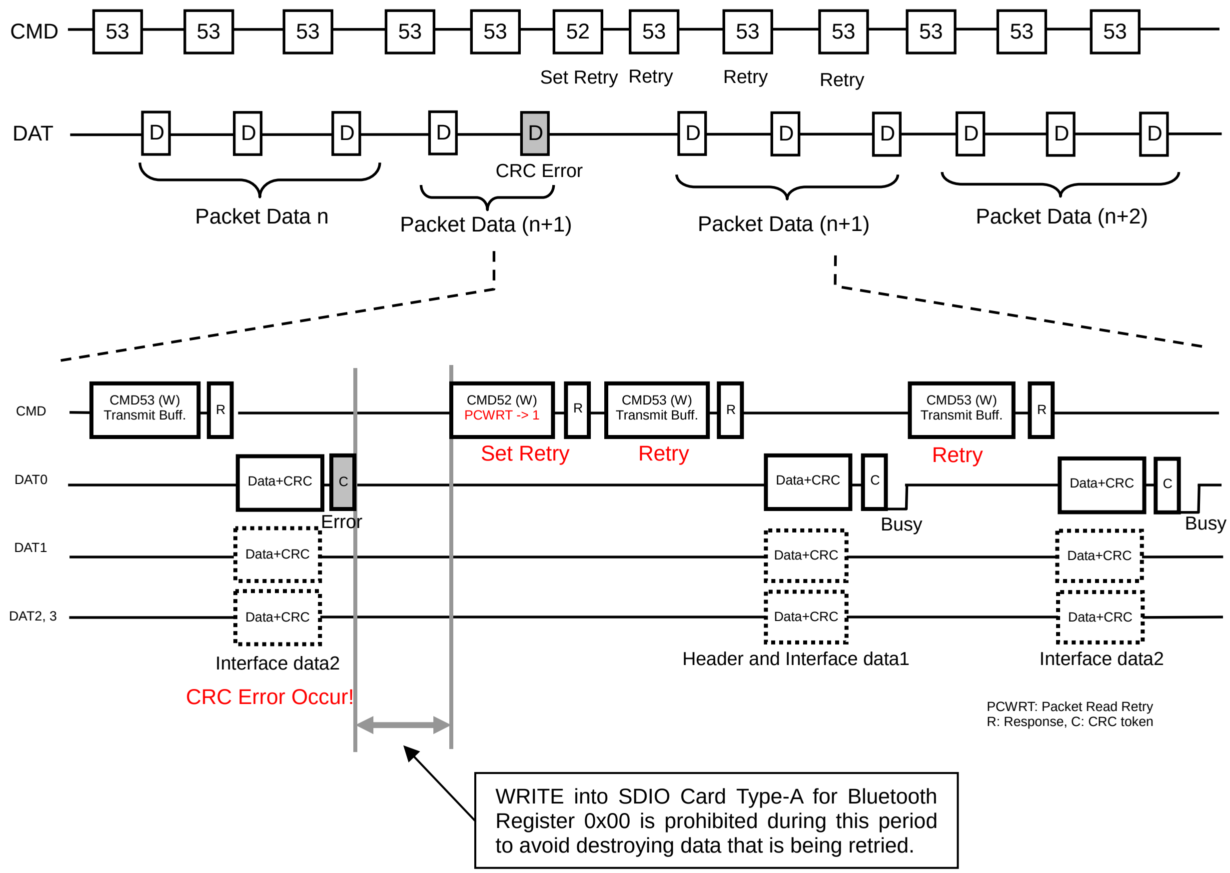

Figure 14 – Write Flow – CRC Error (Byte Basis Transfer Mode)

5.3.3 Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – No Error (Byte Basis Transfer Mode)

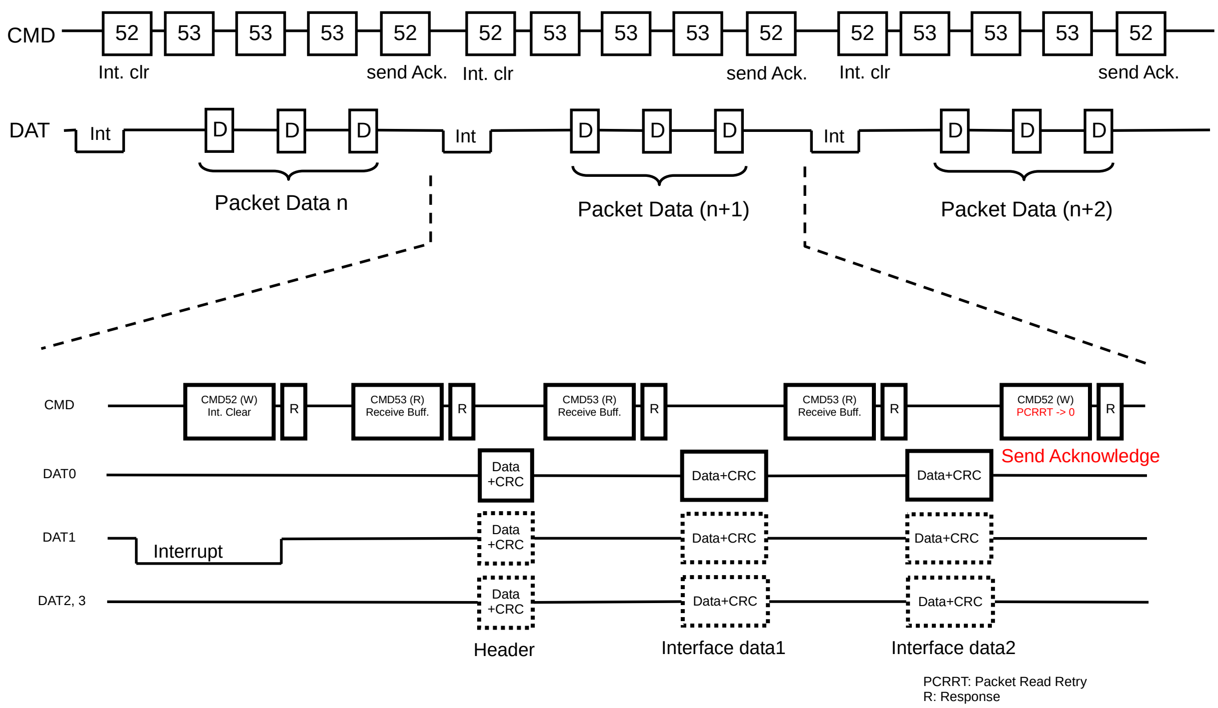

Figure 15 – Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – No Error (Byte Basis Transfer Mode)

5.3.4 Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – No Error (Byte Basis Transfer Mode)

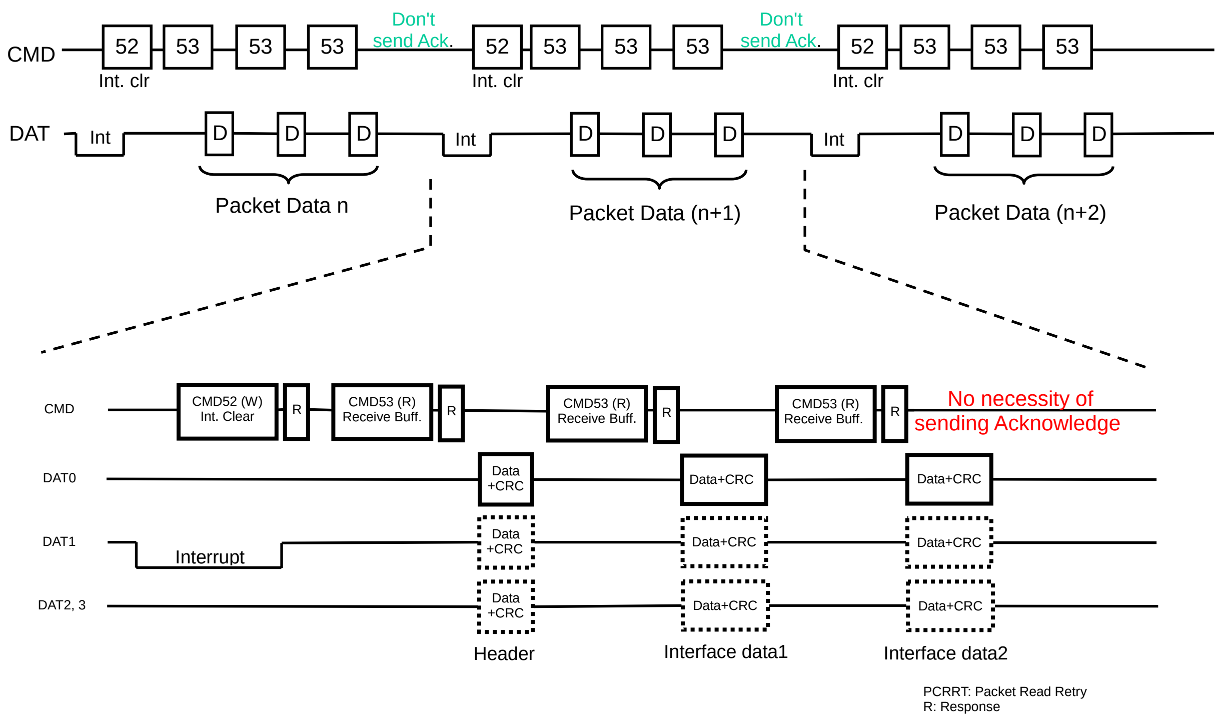

Figure 16 – Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – No Error (Byte Basis Transfer Mode)

5.3.5 Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – CRC Error (Byte Basis Transfer Mode)

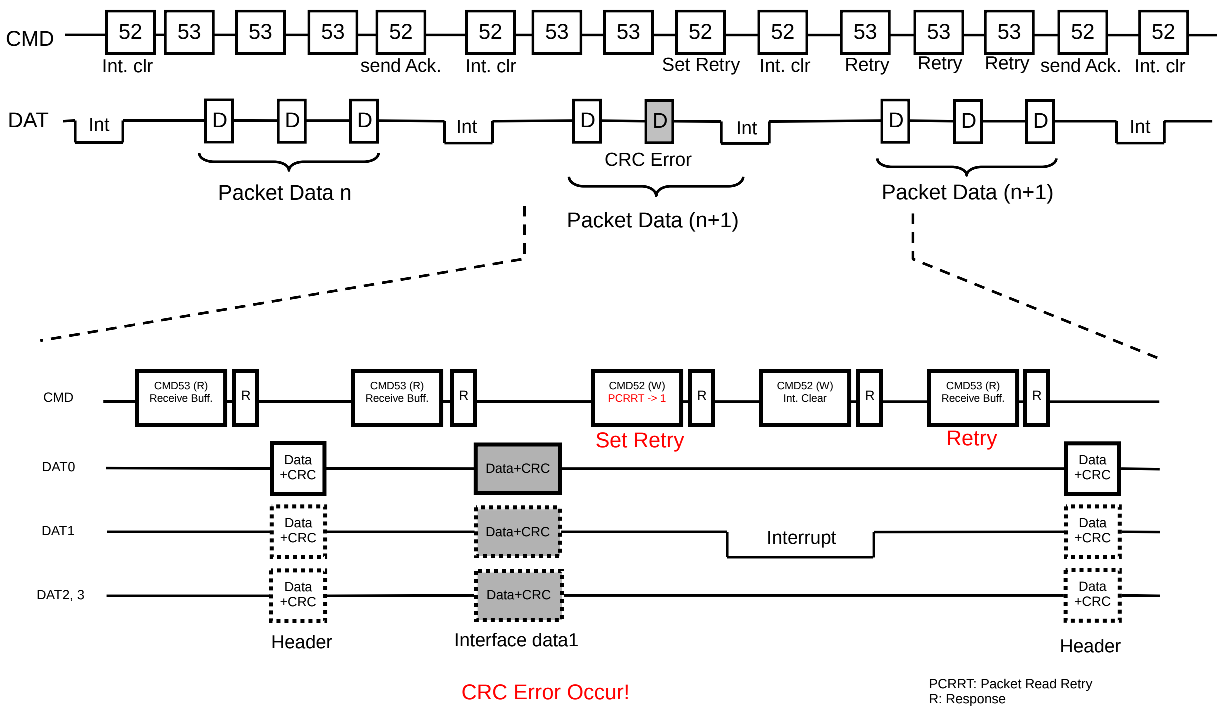

Figure 17 – Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – CRC Error (Byte Basis Transfer Mode)

5.3.6 Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – CRC Error (Byte Basis Transfer Mode)

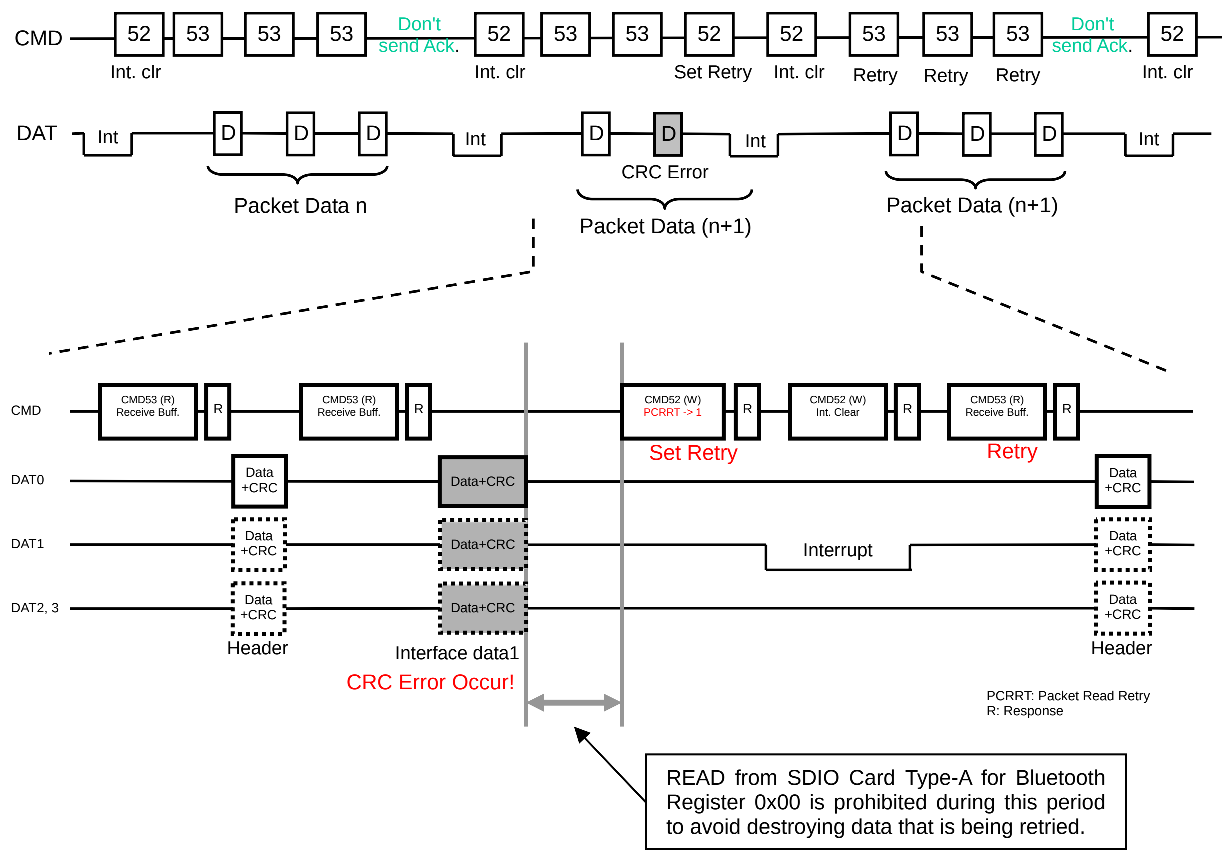

Figure 18 – Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – CRC Error (Byte Basis Transfer Mode)

5.4 Block Basis Transfer Mode

5.4.1 Write Flow – No Error (Block Basis Transfer Mode)

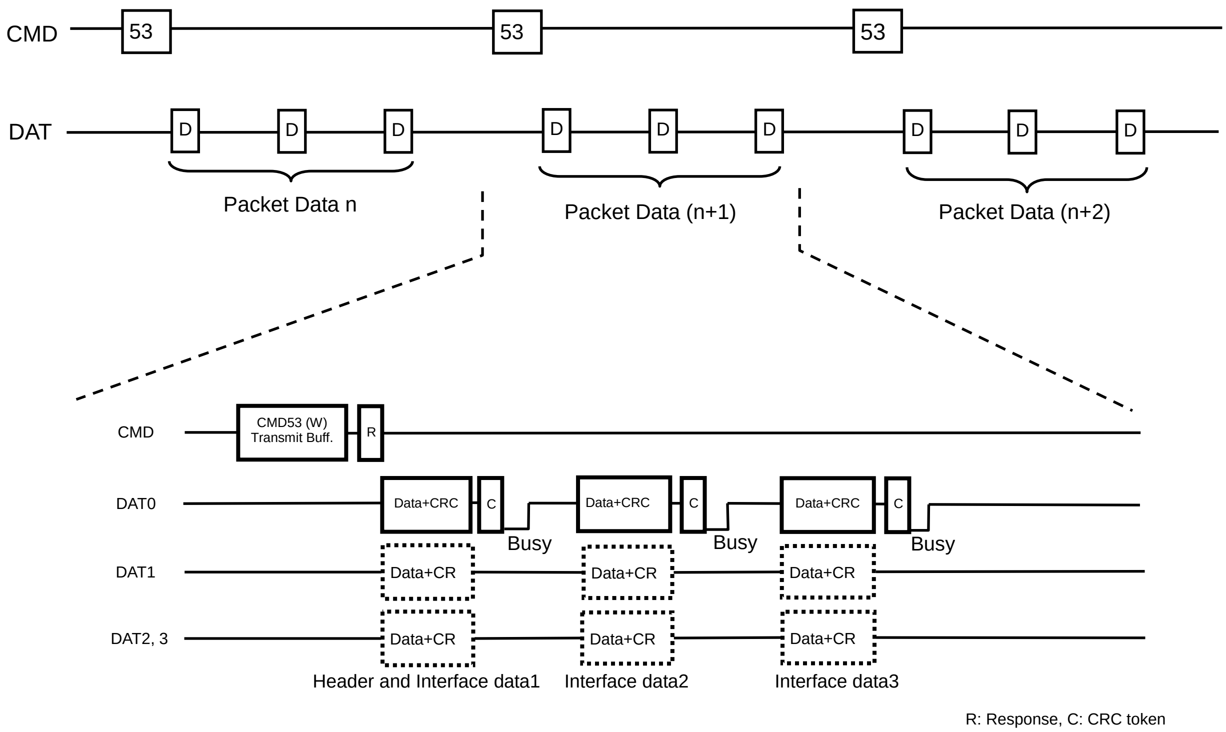

Figure 19 – Write Flow – No Error (Block Basis Transfer Mode)

5.4.2 Write Flow – CRC Error (Block Basis Transfer Mode)

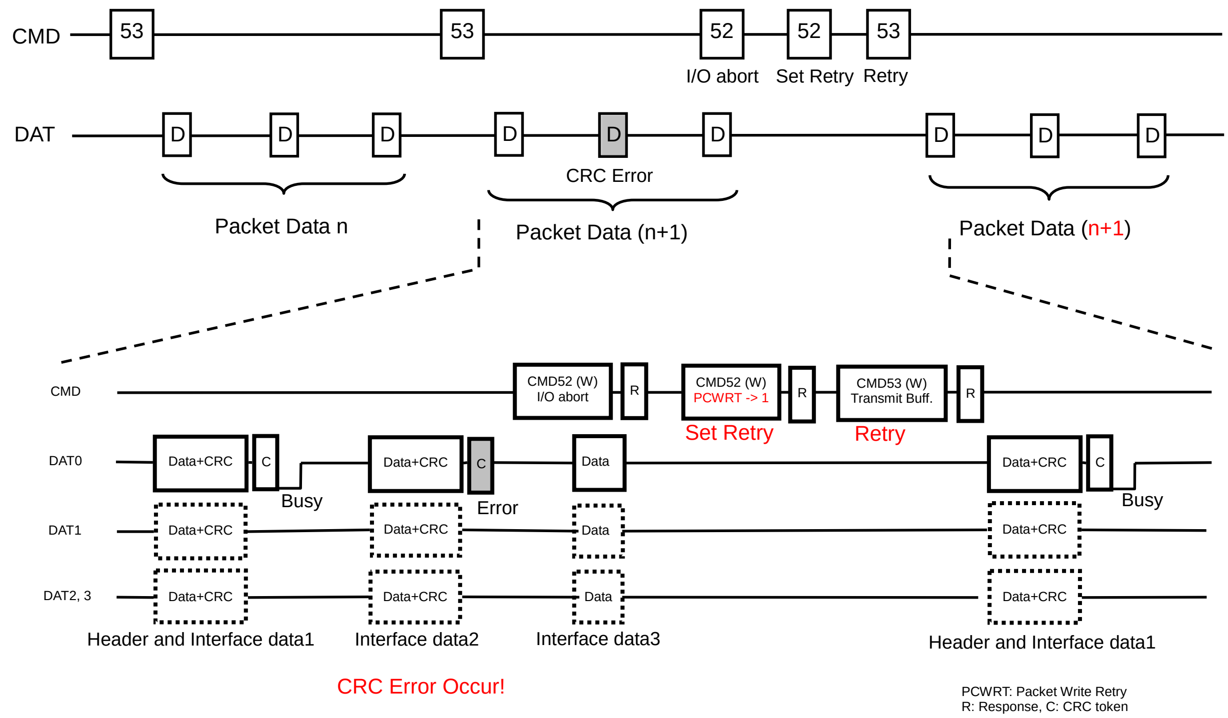

Figure 20 – Write Flow – CRC Error (Block Basis Transfer Mode)

5.4.3 Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – No Error (Block Basis Transfer Mode)

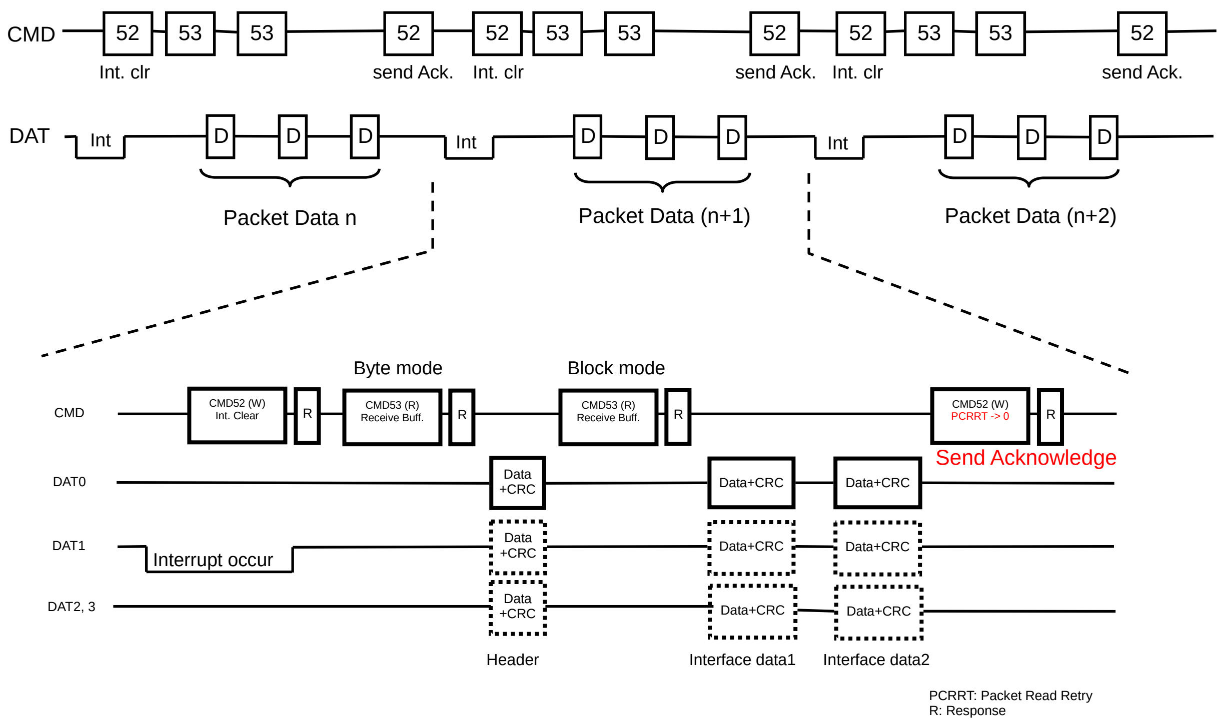

Figure 21 – Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – No Error (Block Basis Transfer Mode)

5.4.4 Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – No Error (Block Basis Transfer Mode)

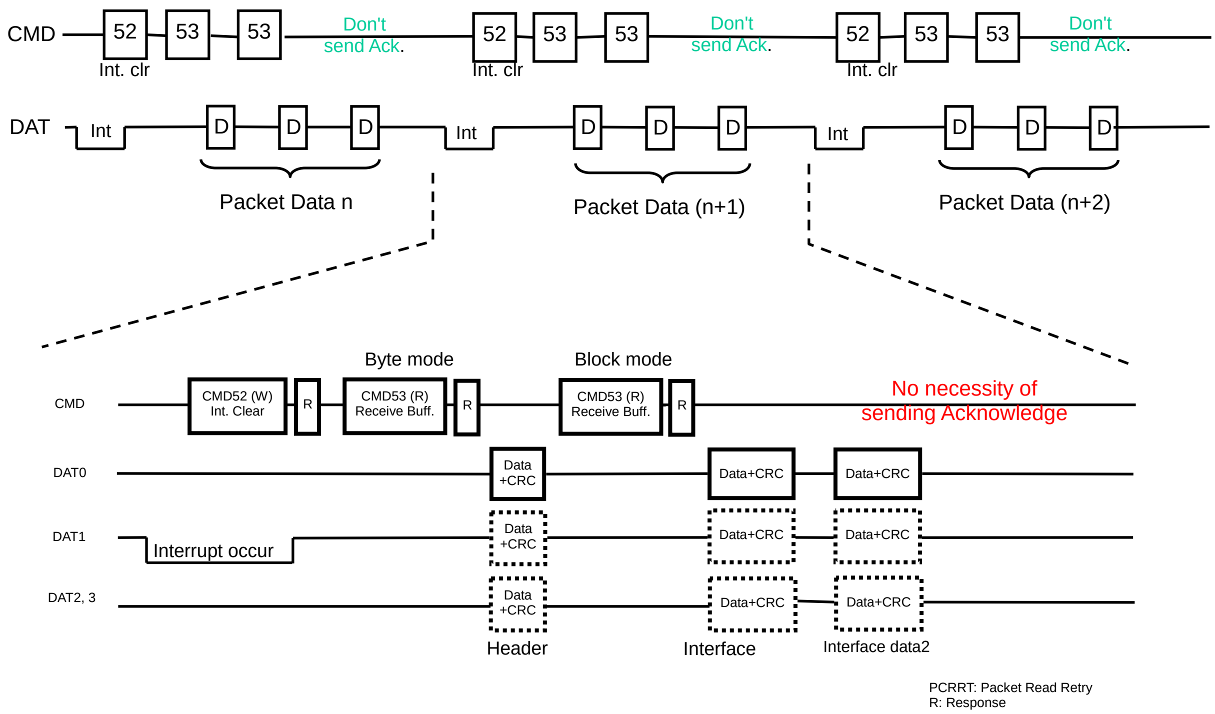

Figure 22 – Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – No Error (Block Basis Transfer Mode)

5.4.5 Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – CRC Error (Block Basis Transfer Mode)

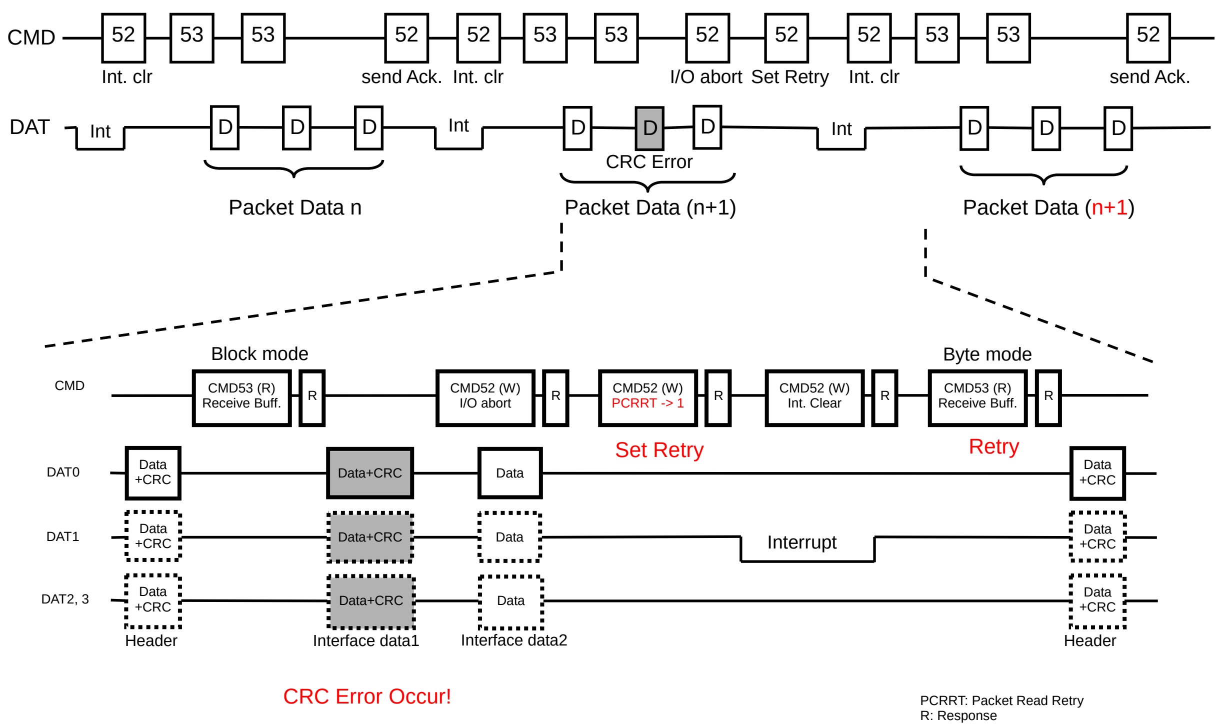

Figure 23 – Read Flow for the Card with Necessary Read Acknowledge (RTC=0) – CRC Error (Block Basis Transfer Mode)

5.4.6 Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – CRC Error (Block Basis Transfer Mode)

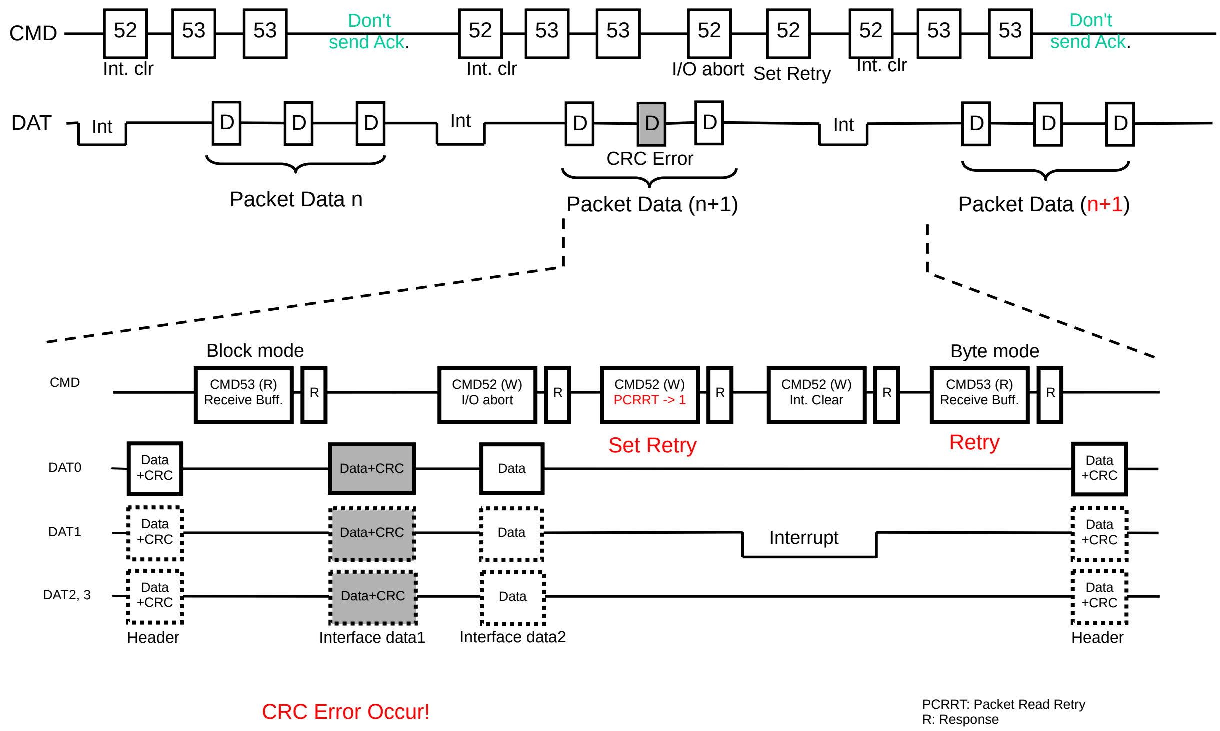

Figure 24 – Read Flow for the Card with Unnecessary Read Acknowledge (RTC=1) – CRC Error (Block Basis Transfer Mode)