Product Data Sheet

ISD-200 ASIC

Part number:
ISD-200 “0002”/”0003”

Revision 2.0

June 15, 2000

Notice:

In-System Design assumes no liability whatsoever for errors and omissions in this document. In-System Design reserves the right to make changes at any time, without notice. Please confirm that this document revision is current.

Table of Contents

Table of Tables

Table of Figures

Document Revision History

Title	ISD-200 ASIC Datasheet
Company	In-System Design Inc. Boise, ID.
Initial Revision #	1.0
Creation Date/Time	March 13, 2000

Revision	Date	Comments
1.0	March 13, 2000	Initial Production Revision
1.1	March 15, 2000	Addition of ATA_EN mode current
1.2	March 17, 2000	Removed reference to 16 byte ATAPI command block support
1.3	March 22, 2000	Added length restriction to USB string descriptors
	April 4, 2000	Added text to indicate a serial number string descriptor is required for MSC compliance
1.4	April 10, 2000	Fixed READ/LOAD_CONFIG parameter descriptions.
1.5	April 26, 2000	Added Internal ROM values to ISD-200 Configuration table. Changed description of some ISD-200 Configuration bits. Restructured the outline to avoid conflicting statements and redundancy. Fixed Table of Tables and Table of Figures Cleaned up and added text on the BUS_POWER pin and related issues.
1.6	April 28, 2000	Made suggested enhancements from review in text.
1.7	May 25, 2000	Several references clarified. First silicon “Rev A” references changed to “first silicon” (ISD-200 “0002” or F731727). Second silicon “Rev B” references changed to “second silicon” (ISD-200 “0003” or F731727A).
1.8	June 13, 2000	Clarify First & Second silicon differences in power management section. Added Second silicon errata. Fixed some typographical errors.
1.9	June 15, 2000	Clarified I_MODE references. Fixed several references for consistency.
2.0	June 29, 2000	Fixed some ATACB descriptions. Added more errata.

Pin Information

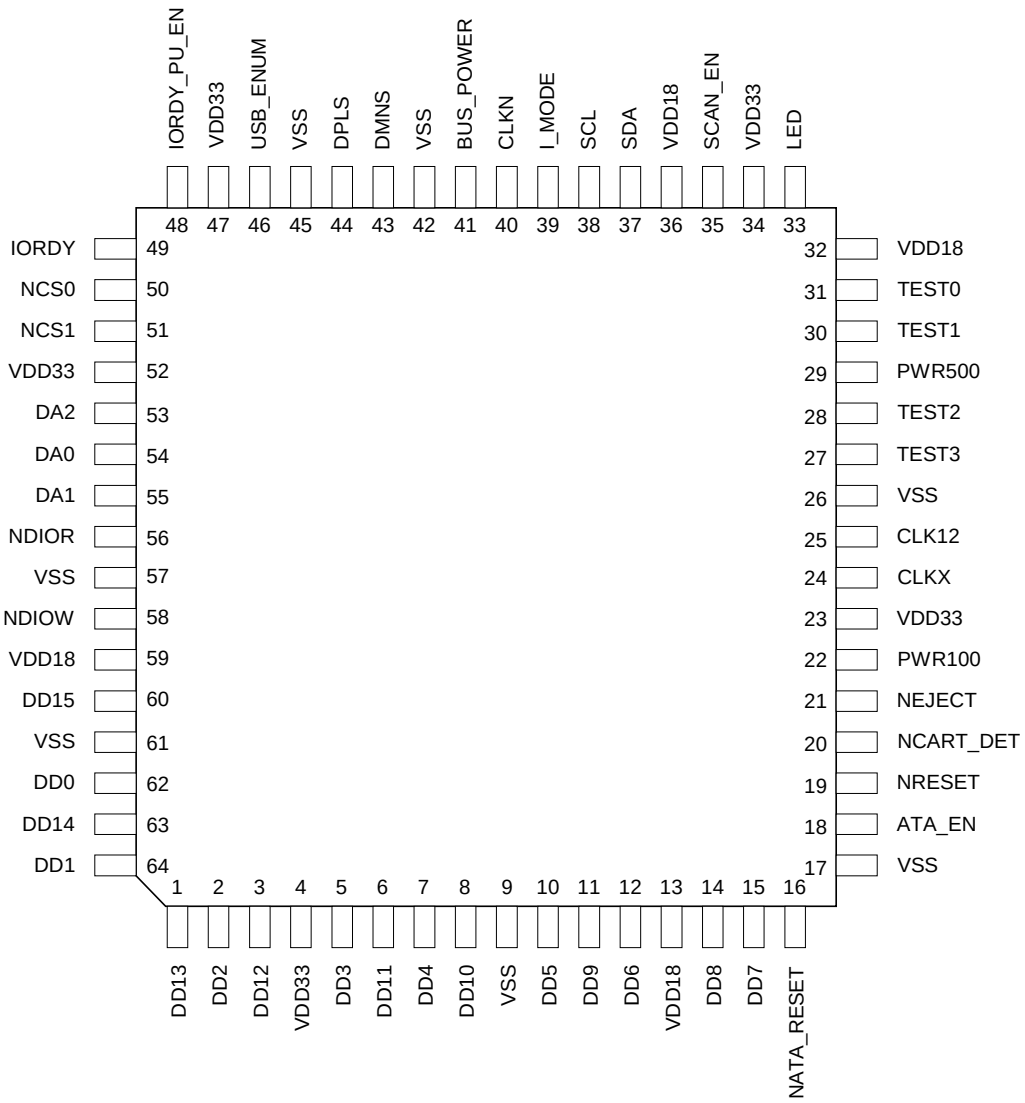


Figure 1 – Pin Layout

Pin Name	TQFP Pin #	Dir	Type	Description
NRESET	19	I	TTL 3.3V	Active low Chip Reset
CLK12, CLKX	25, 24	IO	OSC in/out	Crystal connections. 1.8V tolerant only
TEST0 – TEST3	31, 30, 28, 27	I	TTL 3.3V, pull down	Test mode inputs. 100K internal pull down resistors on TEST(2:0)
SCAN_EN	35	I	TTL 3.3V, pull down	Manufacturing test pin. Scan mode select input. 100K internal pull down resistor.
I_MODE	39	I	TTL 3.3V	USB descriptor and ISD-200 configuration data is obtained from the device by the vendor-specific ATAPI (FBh) command. If set (I_MODE=1) and USB Bus powered (BUS_POWER=1), the device must be able to respond to the FBh ATAPI command when nPWR100 is active. I_MODE operation is not supported in USB Bus powered systems that require more than 100mA of current prior to USB enumeration.
CLKN	40	O	4 mA TTL 3.3V	Configurable clock output. A digital phase lock loop provides a configurable clock source for system integration. (See **Errata #1**)
NEJECT	21	I	TTL 3.3V	Active low media eject request input
NCART_DET	20	I	TTL 3.3V	Active low Cartridge (media) detected input.
ATA_EN	18	I	TTL 3.3V, pull up	ATA interface / USB enable. Allows ATA bus sharing with other host devices. ATA_EN=1 to enable ATA interface. ATA_EN=0 to 3-state to hi-Z ATA interface, disable USB interface. 100K internal pull up resistor (See **Errata #2, #5**)
DA0 – DA2	54, 55, 53	O	4 mA TTL 5V Fail Safe	ATA Address. ATA66 compliant IO cell.
DD0 – DD15	62, 64, 2, 5, 7, 10, 12, 15, 14, 11, 8, 6, 3, 1, 63, 60	IO	4 mA TTL 5V Fail Safe	ATA Data. ATA66 compliant IO cell.
NDIOR	56	O	4 mA TTL 5V Fail Safe	ATA Read Strobe. ATA66 compliant IO cell.
NDIOW	58	O	4 mA TTL 5V Fail Safe	ATA Write Strobe. ATA66 compliant IO cell.
NCS0, NCS1	50, 51	O	4 mA TTL 5V Fail Safe	ATA Chip Selects. ATA66 compliant IO cell.
IORDY	49	I	TTL 5V Fail Safe	ATA Flow Control. ATA66 compliant IO cell.
IORDY_PU_EN	48	O	4 mA TTL 5V Fail Safe	ATA IORDY pull-up connection. Active hi, 3-state to hi-Z off. ATA66 compliant IO cell.
NATA_RESET	16	O	4 mA TTL 5V Fail Safe	ATA pin Reset. ATA66 compliant IO cell.
SCL	38	O	4 mA TTL 5V tolerant	Configuration serial ROM clock. Active low, 3-state to hi-Z off. (See **Errata #4**)
SDA	37	IO	4 mA TTL 5V tolerant	Configuration serial ROM address/data. Active low, 3-state to hi-Z off.
NPWR500	29	O	4 mA TTL 3.3V	Indicates host has enabled use of USB Bus power (USB configuration set to a value other than 0) up to the requested amount in the USB descriptor *bMaxPower* entry. USB Bus powered devices must condition power circuitry with the state of the BUS_POWER signal for correct operation. Active low, 3-state to hi-Z off

Pin Name	TQFP Pin #	Dir	Type	Description
DPLS, DMNS	44, 43	IO	USB Transceiver	USB differential data
USB_ENUM	46	O	4 mA TTL 3.3V	USB enumeration control. Power source / sink for 1.5K pull-up resistor on USB Bus DPLS signal.
LED	33	O	12 mA TTL 3.3V	USB status led control output. Flashes if the USB configuration is set to 0, on solid if USB configuration is set other than 0, and off if USB Bus is suspended. Active low, 3-state to hi-Z off
NPWR100	22	O	4 mA TTL 3.3V	Indicates USB Bus can supply the lesser value of 100 mA or the USB Descriptor *bMaxPower* entry. USB BUS powered devices must condition power circuitry with the state of the BUS_POWER signal for correct operation. Active hi, 3-state to hi-Z off
BUS_POWER	41	I	TTL, pull up	Indicates the source of the ATA / ATAPI device power. (See Errata #6) BUS_POWER=1 indicates USB Bus powered. BUS_POWER=0 indicates self power. 100K internal pull up resistor
VDD18	13, 32, 36, 59			1.8V Core supply
VDD33	4, 23, 47, 52			3.3V IO ring supply
VSS	9, 17, 26, 34, 42, 45, 57, 61			Ground

Table 1 – Pin Descriptions

Overview

- **USB Mass Storage Class Bulk-Only Specification Compliant (Version 1.0 RC)**
- **Command Queuing Hooks In Hardware To Allow Near Theoretical USB Data Transfer Rates (12 Mb/sec)**
- **USB Version 1.1 Compliant**
  - **USB Suspend / Resume Support**
  - **USB Remote Wakeup Support**
- **Integrated USB Transceivers**
- **Two Power Source Modes of Operation**
  - **USB Bus Powered**
  - **Self Powered**
  - **Dynamic (USB Bus or Self Powered) Support with USB Bus Power Capabilities Output Pins**
- **Flexible USB Descriptor and ISD-200 Configuration Retrieval Source**
  - **I2C Serial ROM Interface**
  - **ATA Interface Using Vendor Specific ATA Command (FBh Implemented on ATAPI or ATA Device)**
  - **Default On-Chip ROM Contents**
  - **Support for two Configuration Descriptors for Dynamic (USB Bus or Self) powered applications**
- **Large 32K Byte Data Buffer Maximizes ATA / ATAPI Data Transfer Rate**
- **ATA Interface Supports ATA Modes 0, 1, And 2 Of Operation**
- **IORDY Support**
- **Event Notification Via Vendor Specific ATA Command**
  - **Input Pins For Media Cartridge Detection And Ejection Request**
  - **Vendor Specific ATA Command For Event Notification Is Configurable**
  - **USB Remote Wakeup Support**
- **Multiple LUN support**
- **Full ATA Command Support By Use of Vendor Specific Command Block in the MSC Command Block Wrapper**
- **Configurable External Clock Source (DPLL) To Provide System Level Clock**
- **Provisions To Share ATA Bus With Other Hosts**
- **Uses Inexpensive 12Mhz Crystal For Clock Source**

Introduction

The ISD-200 implements a bridge between one USB 1.1 (http://www.usb.org/developers/data/usbspec.zip) port and one ATA or ATAPI based mass storage device port. This bridge adheres to the *Mass Storage Class Bulk-Only Transport* (http://www.usb.org/developers/data/devclass/usbmassbulk_10.pdf) for ATAPI transactions. Hardware design allows ATAPI command queuing which allows data transfer rates of up to the USB theoretical maximum of 12 Mb/sec.

The USB port of the ISD-200 is connected to a host computer directly or via the downstream port of a USB hub. Host software issues commands and data to the ISD-200 and receives status and data from the ISD-200 using standard USB protocol.

The ATA/ATAPI port of the ISD-200 is connected to a mass storage device. A large 32 Kbyte buffer maximizes ATA/ATAPI data transfer rates by minimizing losses due to device seek times. The ATA interface supports ATA PIO modes 0, 1, and 2.

The device initialization process is configurable, enabling the ISD-200 to initialize most ATA/ATAPI devices without software intervention. The ISD-200 can also be configured to allow software initialization of the device if initialization requirements are not supported by ISD-200 algorithms.

ISD-200 Configuration

Certain timing parameters and operational modes in the ISD-200 are configurable and are controlled by a series of bytes, located at the beginning of the descriptor space.

ISD-200 configuration data is not to be confused with the USB Configuration Descriptor.

ISD-200 Configuration Data and USB Descriptor Sources

ISD-200 configuration data and USB descriptor data in the ISD-200 can be retrieved from several sources, selected by the mode of operation at power up. This provides for maximal flexibility in configuration possibilities. There are two possible external sources for descriptor data. **Table 2** indicates the method of determining which of all data sources are used.

ISD-200 configuration and USB descriptor data can be supplied from an I^2C serial memory device. The ISD-200 can address 4 Kbytes of serial ROM data, but USB descriptor information must be limited to 512 bytes maximum. The ISD-200 provides support for the 24LC01-16 family EEPROM interface.

Alternatively, configuration and descriptor data can be supplied by an attached mass storage device through a vendor-specific Identify (FBh) command. The ISD-200 provides internal RAM (320 bytes) to hold the data.

The ISD-200 also contains an on-board set of ISD-200 configuration and USB descriptors. Retrieval of the on-board data will occur if no external descriptor data is supplied (See **Table 2**). These on-board descriptors may be used during development, prototyping, and manufacturing.

ISD-200 configuration and descriptor sources are selected by the mode of operation detected at power up. The following table describes how the ISD-200 determines USB Descriptor sources.

Serial ROM present	I_MODE Input Pin	ISD-200 Configuration and USB Descriptor Retrieval
No	0	In this mode, the ISD-200 uses a DIE-ID identifier generated at fabrication to provide a unique 12-character Serial Number descriptor string. All other Descriptor and Configuration values are taken from the

Serial ROM present	I_MODE Input Pin	ISD-200 Configuration and USB Descriptor Retrieval
		internal on-board ROM.
No	1	The ISD-200 retrieves all Descriptor and Configuration values from the vendor-specific Identify (FBh) data. The ISD-200 is configured using values in the internal ROM until the FBh data becomes available
Yes	0	The ISD-200 retrieves all Descriptor and Configuration values from serial ROM. The ISD-200 is configured using values present in serial ROM data only (Unless the serial ROM signature check fails. The ISD-200 then uses POR defaults listed in the Descriptor and Configuration data sections of this document).
Yes	1	Undefined mode of operation.

Table 2 – ISD-200 Configuration and Descriptor Sources

Internal ROM Contents
Internal on-board ROM addresses and the contents of those locations are shown in **Tables 3 and 7 to 11**. If the ISD-200 is operated using the on-board ROM contents for descriptor data, the Serial Number string is still guaranteed to be unique, as required by the *USB Mass Storage Class (MSC) Bulk Only Transport Specification*, through the utilization of a DIE-ID process at fabrication. This unique identifier is generated through the usage of physical parameters such as wafer coordinates, wafer number, lot number, fab number, date, time, etc.

The internal ROM is also convenient for prototyping and manufacturing activities. Enough information is supplied from the internal ROM to the host that the ISD-200 will enumerate even when an un-programmed serial ROM is connected.

Serial ROM Interface
The ISD-200 supports the I^2C "fast mode" interface, as found in the 24(L)C01-16 EEPROM family. That device family allows for up to 4 Kbytes of external storage, which may be used to store descriptors and other information. Note that if a 24(L)C04/08/16 part is used, no descriptor is allowed to span multiple pages within the EEPROM.

Programming of the serial ROM can be accomplished using an external device programmer, ISD-200 supported vendor specific USB commands, or using a "bed of nails" while the ISD-200 is in LIMBO Test mode (See **Operational Modes, Test Mode Pins**). An example of Serial ROM formatting is shown in **Appendix A**. Unused space in a serial ROM may be used for other purposes using the LOAD_CONFIG_DATA and READ_CONFIG_DATA USB Requests.

Hardware detects the presence of a serial ROM and, if one is present, performs a data validity check. This check is in the form of a 'signature' and is located at the beginning of the serial ROM data (Addresses 0x0 – 0x1, **Table 3**). If none is detected or the signature check fails, the ISD-200 will return configuration and descriptor data from the internal on-board ROM.

Vendor-Specific Identify (FBh) Data
If a serial ROM is not used, the ISD-200 can be configured to accept Descriptor and Configuration data from an attached device using a vendor-specific Identify command during the boot sequence. (See **Operational Modes, I_MODE Pin**)

For FBh data to be deemed valid, it must first pass a validity check. This check is in the form of a 'signature' and is located at the beginning of the FBh data (Addresses 0x0 – 0x1, **Table 3**). In the event of a failed signature check, the ISD-200 will respond to all Get_Descriptor or Get_Configuration USB

commands by returning the defaults contained in the internal on-board ROM (See **ATA/ATAPI Interface, Vendor-Specific ATA Commands, Identify**).

An example of vendor-specific Identify (FBh) data formatting is shown in **Appendix A**.

ISD-200 Configuration/USB Descriptor Data Formatting

Data formatting for all ISD-200 configuration data and USB descriptor data is identical for Internal ROM, Serial ROM, and vendor-specific Identify (FBh) data. The following sections show how the ISD-200 configuration data is mapped into address space. The **USB Interface** section contains formatting of USB descriptor data (See **Tables 7-11**).

ISD-200 Configuration Data

The ISD-200 Configuration Data is located in addresses 0 to 9 of the Descriptor/Configuration data contents. These bytes are read at power up and determine certain parameters and operational modes used by the ISD-200. Power-on reset default values are specified in bold.

Formatting is identical for the internal ROM, serial ROM, and vendor-specific Identify (FBh) data. See **Appendix A**. The ISD-200 Configuration Bytes get loaded into internal registers, regardless of the original data source.

Address	Field Name	Description	On-board Defaults
0x00	Data Signature (LSB)	This field specifies the least significant byte of the Serial ROM/FBh signature. **This register does not exist in HW (no POR values)**	0x52
0x01	Data Signature (MSB)	This field specifies the most significant byte of the Serial ROM/FBh signature. **This register does not exist in HW (no POR values)**	0x48
0x02	Event Notification	This field specifies the ATA event notification command. Setting this field to 0x00 disables this feature. **POR configuration default of 0x00**	0x00
0x03	DPLL Parameters	This field denotes the parameters used by the internal DPLL. The original clock source is 12 MHz. 5 bits M (7:3), 2 bits N (2:1), 1 bit Enable (0). When enabled, multiply the original clock source by M, divided by N M: 00000 => M=1 00001 => M=1 00010 => M=2 00011 => M=3 …. 11111 => M=31 N: 00 => N=1 01 => N=3 10 => N=2 11 => N=4 Enable: 0 => CLK_N disabled 1 => CLK_N enabled **POR configuration default of 0x00**	0x00
0x04	ATA Initialization Timeout	This field specifies the time in multiples of 128 ms (0x19 = 3.2s) before the ISD-200 stops polling the Alternate Status device register for reset complete and restarts the reset process. NOTE: The ROM contents ATA Initialization Timeout value must be large enough to accommodate I_MODE operation during the first device initialization sequence (before FBh configuration data load) **POR configuration default of 0x02**	0x19

Address	Field Name	Description	On-board Defaults
0x05	Reserved – Bits [7:6] Master/Slave Selection – Bit [5] ATAPI DEVICE RESET – Bit [4] ATA Timing – Bits [3:0]	Reserved; set to '0' This bit specifies device number selection. "0" - Drive 0 "1" - Drive 1 This bit specifies that the ISD-200 perform a ATAPI DEVICE RESET command during a full initialization sequence. This field determines ATA Bus data access cycle times. 0000 reserved 0001 Mode 2 (292 ns) 0010 Mode 2 (333 ns) 0011 Mode 2 (375 ns) 0100 Mode 1 (458 ns) 0101 Mode 1 (500 ns) 0110 Mode 1 (542 ns) 0111 Mode 1 (583 ns) 1000 Mode 0 (625 ns) 1001 Mode 0 (666 ns) 1010 Mode 0 (708 ns) 1011 Mode 0 (750 ns) 1100 reserved 1101 reserved 1110 reserved 1111 reserved **POR configuration default of 0x0B**	0x01
0x06	ATA Command Designator (Byte 0, LSB)	This field specifies the value in CBWCB field that designates if the CB is decoded as ATA commands instead of the ATAPI command block. **POR configuration default of 0x00**	0x24
0x07	ATA Command Designator (Byte 1, MSB)	This field specifies the value in CBWCB field that designates if the CB is decoded as ATA commands instead of the ATAPI command block. **POR configuration default of 0x00**	0x24

Address	Field Name	Description	On-board Defaults
0x08	Initialization Status – Bit[7]	This (Read Only) bit denotes the Device Initialization Status. If set, indicates the device initialization sequence is active.	0x08
	Configuration Descriptor 2 Present – Bit[6]	This bit specifies that a second Configuration Descriptor is present. See "BUS_POWER pin" under the "Power Management" section. "0" - not present "1" - present	
	Skip ATA / ATAPI Device Initialization – Bit[5]	This bit specifies that the ISD-200 skip device initialization. "0" - normal operation "1" - only reset the device prior to allowing USB enumeration. The function of this bit is overridden (0) if I_MODE is set. Notification command, descriptor override, SRST enable, ATAPI DEVICE RESET settings are irrelevant.	
	ATA HIGH POWER Device for USB Bus Powered Devices – Bit[4]	Setting this bit indicates that when USB Bus powered, the system's ATA interface is only powered when the NPWR500 pin is active (not controlled by NPWR100). "0" – normal operation "1" – If (BUS_POWER = 1) and (USB configuration = 0) then ATA pads are put into low power mode, ATA control lines (RESET-, DIOR-, DIOW-) are 3-stated to hi-Z, and the remainder of the ATA lines(NCS(1:0), DA(2:0), DD(15:0), and IORDY_PU_EN) are driven low. (See **Power Management** section)	
	Descriptor Override – Bit[3]	Setting this bit causes the ISD-200 to override the USB Descriptors for ATA devices. Not compatible with setting BUS_POWER=1. If set, (I_MODE = 0), AND (ATA device identified or device initialization fails), bDeviceClass returned as 0xFF bDeviceSubClass returned as 0x00 bDeviceProtocol returned as 0xFF bInterfaceClass returned as 0xFF bInterfaceSubClass returned as 0x00 bInterfaceProtocol returned as 0xFF idProduct bit 0 is replaced with a '1' (Suggested: even idProduct for ATAPI devices, odd numbers for ATA devices) else USB descriptor information unaltered	
	Last LUN Identifier – Bits[2:0]	This field denotes the (zero-based) maximum number of LUNs supported. **POR configuration default of 0x30**	
0x09	Reserved – Bits[7:2]	Reserved; set to '0'	0x01
	Report One Configuration – Bit[1]	This bit determines whether one or two configuration descriptors are made available to the host. This bit should not be set if the Configuration Descriptor 2 Present configuration bit is not set. "0" - Both descriptors are reported to the host if BUS_POWER=0 "1" - Report only the configuration descriptor that corresponds to state of the BUS_POWER input. Configuration 1 is returned when BUS_POWER=1, configuration 2 is returned when BUS_POWER=0	
	SRST Enable – Bit[0]	Setting this bit enables the SRST reset algorithm in the ISD-200. **POR configuration default of 0x00**	

Table 3 – ISD-200 Configuration Bytes

Valid Configuration Byte Settings

The following table depicts valid combinations of configuration byte settings and inputs that depend on the configuration source and USB Bus power usage. Operation outside the defined values may result in unexpected behavior and should be avoided.

Mode of Operation	I_MODE (input pin)	Descriptor Override (configuration bit)	ATA HIGH POWER (configuration bit)	Configuration Descriptor 2 Present (configuration bit)	Report One USB Configuration (configuration bit)
Configuration Source					
Serial ROM	0	0/1	0/1		
FBh (I_MODE) Data	1	0	0		
Internal ROM	0	0/1	0/1		
Attached Device Power Source					
USB Bus Only		0	0/1	0	0
Self-Powered Only		0/1	0	0	0
USB Bus or Self Powered		0	0/1	0	0
USB Bus or Self Powered		0	0/1	1	0/1
Note: Internal ROM value options are listed here in case the Internal ROM mask is rolled for a specific product/vendor application. See **Table 3 and 7-11** for Internal ROM contents.					

Table 4 – Valid Configuration Byte Settings

USB Interface

The USB port on the ISD-200 is electrically and logically compliant with the *Universal Serial Bus Specification Revision 1.1 (*http://www.usb.org/developers/data/usbspec.zip).

Descriptor Requirements

Descriptors programmed into a serial ROM or vendor-specific Identify (FBh) command data must observe the following constraints.

Configuration Descriptors

The ISD-200 can support one or two configuration descriptors. If the system can operate as USB Bus powered and self powered, the ISD-200 requires that the first configuration descriptor be “dynamic” and the second be self powered. If the system can only operate as USB Bus powered or self powered, then only the first configuration descriptor is used (the second configuration descriptor is zeroed out and not reported to the host). See **Power Management , BUS_POWER pin** section.

String Descriptor Indexes

The ISD-200 imposes constraints on what descriptor string index values are allowable. For those strings that are optional, an index of 0x00 indicates that string is absent. Allowable string indexes are as follows:

iManufacturer -	0x01 or 0x00 if unused
iProduct -	0x02 or 0x00 if unused
iConfiguration1 -	0x03 or 0x00 if unused
iInterface1 -	0x04 or 0x00 if unused
iSerialNumber -	0x05 (must be present to be *MSC* compliant)
iConfiguration2 -	0x07 or 0x00 if unused
iInterface2 -	0x08 or 0x00 if unused

Descriptor Override
If the ISD-200 is to be used in applications where either ATA or ATAPI devices may be connected, the Descriptor Override configuration bit must be set. To avoid confusion, it is suggested that the idProduct(LSB) descriptor field be assigned an even or odd value depending on the device type. The detection of an ATA or ATAPI device initialization failure results in bit-0 of this value being replaced with '1'. Thus, it is suggested that ATAPI devices should have even idProduct values and ATA devices odd.

Pipes

This ISD-200 provides four USB pipes: Default Control, Bulk Out, Bulk In, and Interrupt.

Default Control Pipe
The default pipe is used to transport standard, class and vendor-specific USB requests to the ISD-200.

Bulk Out Pipe
The Bulk Out pipe is used to send command and data to an attached mass storage device. Maximum packet size is 64 bytes.

Bulk In Pipe
The Bulk In pipe is used to receive status and read data from an attached mass storage device. Maximum packet size is 64 bytes.

Interrupt Pipe
The Interrupt pipe is implemented for legacy driver compatibility reasons only. If addressed, it will always return 0x00.

Requests

The ISD-200 responds to three different types of request:
- Standard USB device requests
- Mass Storage Class Bulk-Only requests
- Vendor-specific requests

Standard Requests
The ISD-200 supports all USB standard device requests except the optional Set Descriptor request. These requests, which are described in Chapter 9, Device Framework, of the *USB Specification*, are:
- Clear Feature
- Get Configuration
- Get Descriptor (for information on String Descriptors, see **String Descriptors** on page 23)
- Get Interface
- Get Status
- Set Address
- Set Configuration
- Set Interface
- Set Feature

Mass Storage Class Bulk-Only Requests
Mass Storage Class Bulk-Only requests supported by the ISD-200 are listed in the following table.

Label	bmRequestType	bRequest	wValue	wIndex	wLength	Data
HARD_RESET	00100001b	11111111b	0000h	Interface	0000h	[None]
GET_MAX_LUN	10100001b	11111110b	0000h	Interface	0001h	1 byte

Table 5 – Mass Storage Class Bulk-Only Requests

HARD_RESET
This request flushes all buffers and resets the pipes to their default states, resets all hardware and registers to their default state, causes the ISD-200 to enter a power-up reset state, and resets the attached ATA device with a pin reset (NATA_RESET). Any STALL conditions or bulk data toggle bits remain unchanged.

Vendor-Specific Requests
Vendor specific requests supported by the ISD-200 are listed in the following table.

Label	bmRequestType	bRequest	wValue	wIndex	WLength	Data
LOAD_CONFIG_DATA	01000000b	00000001b	Data Source	Starting Address	Data Length	Write Data
READ_CONFIG_DATA	11000000b	00000010b	Data Source	Starting Address	Data Length	Read Data
SOFT_RESET	01000000b	00000011b	0000h	0000h	0000h	[None]
CMD_QUEUING_CONTROL	01000000b	00000100b	0000h	Queuing Control	0000h	[None]

Table 6 – Vendor-Specific Requests

LOAD_CONFIG_DATA
This request allows configuration data to be written to the data source specified by the wValue field. The wIndex field specifies the starting address in the data source to which data is to be written and the wLength field denotes the length in bytes of data to be written.

Legal values for wValue are as follows:

0x0000	Configuration bytes, addresses 0x2 – 0x9 only
0x0002	External serial ROM

Writes to serial ROM may only be written starting on eight-byte boundaries, which means that the address value must be evenly divisible by eight. Also, writes to the serial ROM may only include a single 256 byte page per transaction. For example: 250 bytes may be written if the Starting Address is 6. (See **Errata #3**)

Writes to the configuration bytes must be constrained to addresses 0x2 through 0x9, discussed in the ISD-200 Configuration data section (See **Table 3**). Attempts to write outside this address space will result in a STALL condition. Only ISD-200 Configuration Byte registers get over written and not the original data source (serial ROM, FBh data, or on-board ROM).

Illegal values for wValue as well as attempts to write to a serial ROM when none is connected will result in a STALL condition on the USB port.

READ_CONFIG_DATA
This USB request allows data to be retrieved from the data source specified by the wValue field. Data is retrieved from the data source, beginning at the address specified by wIndex. The wLength field denotes the length in bytes of data to be read from the data source.

Legal values for wValue are as follows:

0x0000	Configuration bytes, addresses 0x2 – 0x9 only
0x0001	Internal on-board ROM
0x0002	External serial ROM
0x0003	Vendor-specific Identify (FBh) data

Illegal values for wValue will result in a STALL condition on the USB port. Attempted reads from a serial ROM when none is connected or attempted reads from FBh data when not in I_MODE or when a serial ROM is present will result in a STALL condition. Attempts to read configuration bytes outside the address space 0x2 – 0x9 will also result in a STALL condition.

SOFT_RESET
This request resets the ISD-200 data path control state machines, buffer ram and the command queue. The attached device does not get reset.

This USB request is required for error recovery if complex command queuing is used.

CMD_QUEUING_CONTROL
This request sets the type of command queuing used by the ISD-200. Enabling command queuing allows the ISD-200 to accept a CBW (refer to the *USB Mass Storage Class Bulk Only Transport Specification*) from a new command before the CSW for an earlier command has been sent. There are two types of command queuing, simple command queuing and complex command queuing.

Simple command queuing allows the ISD-200 to accept a CBW for a new command before the previous command completes. If the previous command is a bulk out transfer, the CBW is not taken until all data for the previous command has been transferred. The new command will not be operated on until the CSW for the prior command has been sent. Error recovery is done through a HARD_RESET request.

Complex command queuing enables out of order error recovery without resetting the attached device. Command queuing is done in the same manner as in simple command queuing. However, in the event of a an error, the ISD-200 can now accept a SOFT_RESET which resets the ISD-200 but does not reset the device.

Legal values for wIndex are as follows:

0x0000	Simple Command Queuing (**POR default**)
0x0001	Complex Command Queuing

Descriptors

Supported Descriptors

- *Device*
- *Configuration*
  The ISD-200 supports one or two possible configurations depending on the mode of operation. See **Power Management, BUS_POWER pin** section.
  Configuration 1. This configuration descriptor is used for Self, USB Bus, or Dynamic power applications. "Dynamic" power is defined to mean that the ISD-200 can operate as Self or USB Bus powered. The maximum power that can be drawn from the USB Bus is specified in the bMaxPower descriptor field.

Configuration 2. This configuration denotes that the ISD-200 is Self powered only and is only used if Configuration 1 is Dynamic power.

- *Interface*
  The ISD-200 supports one interface with four possible endpoints.
  - *Endpoint*
    The ISD-200 supports the following endpoints:
    Default Control endpoint. Accessible as endpoint 0.
    Bulk Out endpoint. Accessible as endpoint 1.
    Bulk In endpoint. Accessible as endpoint 2.
    Interrupt endpoint. Accessible as endpoint 3.
  - *String*
    The ISD-200 supports a set of class and vendor-specific string descriptors. For more information on strings, refer to the following section.

Descriptor Data Format

Device Descriptor
There is only one device descriptor for each USB device. This descriptor gives USB information about the ISD-200 device such as definitions of the device class and device subclass, among other things.

The bNumConfigurations field specifies how many configurations the ISD-200 supports. See **Power Management, BUS_POWER Pin section**.

Address	Field Name	Description	On-board Defaults
0x12	bLength	Length of device descriptor in bytes.	0x12
0x13	bDescriptor Type	Descriptor type.	0x01
0x14	bcdUSB (LSB)	USB Specification release number in BCD.	0x10
0x15	bcdUSB (MSB)		0x01
0x16	bDeviceClass	Device class. If (Descriptor Override configuration bit = 1 and I_MODE = 0) AND (an ATA device is detected or device initialization fails to complete), this field is returned as 0xFF.	0x00
0x17	bDeviceSubClass	Device subclass. If (Descriptor Override configuration bit = 1 and I_MODE = 0) AND (an ATA device is detected or device initialization fails to complete), this field is returned as 0x00	0x00
0x18	bDeviceProtocol	Device protocol. If (Descriptor Override configuration bit = 1 and I_MODE = 0) AND (an ATA device is detected or device initialization fails to complete), this field is returned as 0xFF	0x00
0x19	bMaxPacketSize	Maximum USB packet size supported	0x40
0x1A	idVendor (LSB)	Vendor ID.	0xAB
0x1B	idVendor (MSB)		0x05
0x1C	idProduct (LSB)	Product ID. If (Descriptor Override configuration bit = 1 and I_MODE = 0) AND (an ATA device is detected or device initialization fails to complete), this field is returned as idProduct bit 0 is replaced with a '1'	0x30
0x1D	idProduct (MSB)		0x00
0x1E	bcdDevice (LSB)	Device release number in BCD.	* 0x00 ** 0x10
0x1F	bcdDevice (MSB)		0x01
0x20	iManufacturer	Index to manufacturer string. This entry must be set to 0x01 if string is present, else 0x00 if not present	0x01

Address	Field Name	Description	On-board Defaults
0x21	iProduct	Index to product string. This entry must be set to 0x02 if string is present, else 0x00 if not present	0x02
0x22	iSerialNumber	Index to serial number string. This entry must be set to 0x05 if string is present, else 0x00 if not present. The *USB Mass Storage Class Bulk Only Transport Specification* requires a unique serial number.	0x05
0x23	bNumConfigurations	Number of configurations supported. This value must be consistent with configuration settings for 'Configuration Descriptor 2 Present' and 'Report 1 Configuration'. Valid values are: 1 (CFG_2_PRESENT=0) OR (CFG_2_PRESENT=1 AND RPT_1_CFG=1) 2 (CFG_2_PRESENT=1 AND RPT_1_CFG=0) If (`Descriptor 2 Present' = 1) and ('Report 1 Configuration' = 0) and (BUS_POWER = 1) then this value is overridden with 0x01.	0x01
* - First Silicon value "0002" ** - Second Silicon value "String(6) =			

Table 7– Device Descriptor

Configuration Descriptor

The ISD-200 supports up to two configuration descriptors. The configuration descriptor contains information about the ISD-200 device configuration. Each configuration has one interface that supports four endpoints.

If a second configuration descriptor is used, offset locations 0x24-0x27 in the following table are used to override the appropriate values. The common descriptor fields are shared with the first configuration. See **Power Management, BUS_POWER pin** section.

Address	Field Name	Description	On-board Defaults
0x24	bConfiguration Value 2	The value to use as an argument to Set Configuration to select the configuration. This entry must be set to 0x02 if two configuration descriptors are present.	0x00
0x25	iConfiguration 2	Index to second configuration string. This entry must be set to 0x07 if string is present, otherwise set to 0x00 if not present.	0x00
0x26	bMaxPower 2	Maximum power consumption for the second configuration. Units used are mA*2 (i.e. 0x31 = 98 mA).	0x00
0x27	iInterface 2	Index to interface string associated with the second configuration descriptor. This entry must be set to 0x08 if string is present, otherwise set to 0x00 if not present.	0x00
0x28	bLength	Length of configuration descriptor in bytes.	0x09
0x29	bDescriptorType	Descriptor type.	0x02
0x2A	bTotalLength (LSB)	Number of bytes returned in this configuration. This includes the configuration descriptor plus all the interface and endpoint descriptors.	0x27
0x2B	bTotalLength (MSB)		0x00
0x2C	bNumInterfaces	Number of interfaces supported. The ISD-200 only supports one interface.	0x01
0x2D	bConfiguration Value 1	The value to use as an argument to Set Configuration to select the configuration. Set to 0x01 for the first configuration descriptor.	0x01
0x2E	iConfiguration 1	Index to first configuration string. This entry must be set to 0x03 if string is present, otherwise set to 0x00 if not present.	0x00
0x2F	bmAttributes	Device attributes for this configuration. Configuration characteristics: Bit Description On-board default 7 Reserved. '1' 6 Self-powered. '1' 5 Remote wake-up. '1'/'0' (See **Operational Modes, NEJECT, NCART_DET Pins**) 4-0 Reserved, set to 0. '0'	* 0xE0 ** 0xC0
0x30	bMaxPower 1	Maximum power consumption for the second configuration. Units used are mA*2 (i.e. 0x31 = 98 mA).	0x31

Address	Field Name	Description	On-board Defaults
* - First Silicon value ** - Second Silicon value			

Table 8 – Configuration Descriptor(s)

Interface Descriptor
This descriptor specifies the specific interface within a configuration. In the ISD-200, the interface contains four endpoint descriptors: Default Control (no descriptor), Bulk out, Bulk in, and Interrupt. It should be noted that the Interrupt endpoint is only present for driver legacy reasons. It shall always return 0x00 when it is polled.

Interface and endpoint descriptors cannot be directly accessed using the Get_Descriptor USB command. However, interface and endpoint descriptors are always returned or written to as part of the configuration descriptor.

Endpoint descriptors and addresses must be in the fixed order of the ISD-200 on-board defaults. Bulk-out first, then Bulk-in followed by Interrupt.

Address	Field Name	Description	On-board Defaults
0x31	bLength	Length of interface descriptor in bytes.	0x09
0x32	bDescriptorType	Descriptor type.	0x04
0x33	bInterfaceNumber	Interface number.	0x00
0x34	bAlternateSettings	Alternate settings	0x00
0x35	bNumEndpoints	Number of endpoints	0x03
0x36	bInterfaceClass	Interface class. If (Descriptor Override configuration bit = 1 and I_MODE = 0) AND (an ATA device is detected or device initialization fails to complete), this field is returned as 0xFF.	0x08
0x37	bInterfaceSubClass	Interface subclass. If (Descriptor Override configuration bit = 1 and I_MODE = 0) AND (an ATA device is detected or device initialization fails to complete), this field is returned as 0x00.	0x06
0x38	bInterfaceProtocol	Interface protocol. If (Descriptor Override configuration bit = 1 and I_MODE = 0) AND (an ATA device is detected or device initialization fails to complete), this field is returned as 0xFF.	0x50
0x39	iInterface 1	Index to interface string associated with the first configuration descriptor. This entry must be set to 0x04 if string is present, otherwise set to 0x00 if not present.	0x00
USB Bulk Out Endpoint			
0x3A	bLength	Length of this descriptor in bytes.	0x07
0x3B	bDescriptorType	Endpoint descriptor type.	0x05
0x3C	bEndpointAddress	This is an Out endpoint, endpoint number 1.	0x01
0x3D	bmAttributes	This is a bulk endpoint.	0x02
0x3E	wMaxPacketSize (LSB)	Max data transfer size.	0x40
0x3F	wMaxPacketSize (MSB)		0x00
0x40	bInterval	Does not apply to bulk endpoints.	0x00
USB Bulk In Endpoint			
0x41	bLength	Length of this descriptor in bytes.	0x07
0x42	bDescriptorType	Endpoint descriptor type.	0x05
0x43	bEndpointAddress	This is an In endpoint, endpoint number 2.	0x82
0x44	bmAttributes	This is a bulk endpoint.	0x02

Address	Field Name	Description	On-board Defaults
0x45	wMaxPacketSize (LSB)	Max data transfer size.	0x40
0x46	wMaxPacketSize (MSB)		0x00
0x47	bInterval	Does not apply to bulk endpoints.	0x00
USB Interrupt Endpoint			
0x48	bLength	Length of this descriptor in bytes.	0x07
0x49	bDescriptorType	Endpoint descriptor type.	0x05
0x4A	bEndpointAddress	This is an Interrupt endpoint, endpoint number 3.	0x83
0x4B	bmAttributes	This is an interrupt endpoint.	0x03
0x4C	wMaxPacketSize (LSB)	Max data transfer size.	0x02
0x4D	wMaxPacketSize (MSB)		0x00
0x4E	bInterval	This is the polling interval.	0x20

Table 9 – Interface Descriptor

String Descriptors

The ISD-200 supports 9 USB string descriptors. These strings can be referenced by standard descriptors (e.g. a manufacturer name string indexed by the iManufacturer field in the Device Descriptor). *All string descriptor lengths are restricted to 63 bytes or less*.

All optional string descriptors are referenced using a table of starting string addresses (See **Table 10**). The starting address values are specified as the descriptor location divided by 2 (e.g. Language ID string begins at 0x50 but is specified in the table to be 0x28). If a particular string isn't implemented, the starting address value must be set to 0x00. String index 6 is hard coded to return the English Unicode ISD-200 Hardware revision string "0002" or "0003".

Similarly, as specified in the descriptor tables, if a string isn't implemented, the index (example: iProduct in the device descriptor) reference must be set to 0x00.

Address	Field Name	Description	On-board Defaults
0x0A	Language ID String(0) Starting Address	Address location for LANGID string (divided by 2). This offset must be set to 0x00 if the string is not present.	0x28
0x0B	Manufacturer String(1) Starting Address	Address location for iManufacturer string (divided by 2). This offset must be set to 0x00 if the string is not present.	0x2A
0x0C	Product String Starting(2) Address	Address location for iProduct string (divided by 2). This offset must be set to 0x00 if the string is not present.	0x3B
0x0D	Configuration 1 String(3) Starting Address	Address location for iConfiguration string (divided by 2). This offset must be set to 0x00 if the string is not present.	0x00
0x0E	Configuration 1 Interface String(4) Starting Address	Address location for iInterface string (divided by 2). This offset must be set to 0x00 if the string is not present.	0x00
0x0F	Serial Number String(5) Starting Address	Address location for iSerialNumber string (divided by 2). This offset must be set to 0x00 if the string is not present. The *USB Mass Storage Class Bulk Only Transport Specification* requires a unique serial number.	0x00
0x10	Configuration 2 String(7) Starting Address	Address location for second iConfiguration 2 string (divided by 2). This offset must be set to 0x00 if the string is not present.	0x00
0x11	Configuration 2 Interface String(8) Starting Address	Address location for second iInterface 2 string (divided by 2). This offset must be set to 0x00 if the string is not present.	0x00

Table 10 – String Locations

String index 0 must contain the LANGID of exactly one language, as the ISD-200 supports only a single language. Microsoft defines the LANGID codes for Windows, as described in *Developing International Software for Windows 95 and Windows NT*, Nadine Kano, Microsoft Press, Redmond, Washington. Note that the LANGID code for English is 0x0409 (http://www.usb.org/developers/data/USB_LANGIDs.pdf).

There is an additional string, index 6, that is not referenced by standard descriptors. Index 6 contains the ISD-200 Hardware Revision string.

The following table shows how the LANGID, manufacturer, and product strings are formatted in the on-board ROM contents, and this can be considered an example of how to format strings in a serial ROM or in FBh data. Each string character is comprised of an ASCII character appended to a NULL byte to meet the UNICODE encoding requirements as specified in *The Unicode Standard, Worldwide Character Encoding, Version 1.0, Volumes 1 and 2.*

Address	Field Name	Description	On-board Defaults
USB String Descriptor - Index 0 (LANGID)			
0x50	bLength	LANGID string descriptor length in bytes.	0x04
0x51	bDescriptorType	Descriptor type.	0x03
0x52	LANGID (LSB)	Language supported. Note: See LANGID table in Microsoft documentation (the code for English is 0x0409)	0x09
0x53	LANGID (MSB)		0x04
USB String Descriptor - Index 1 (Manufacturer)			
0x54	bLength	String descriptor length in bytes (restricted to 63 bytes or less).	0x22
0x55	bDescriptorType	Descriptor type.	0x03
0x56	bString	ASCII character.	0x49 (“I”)
0x57	bString	(“NUL”)	0x00
0x58	bString	ASCII character.	0x6E (“n”)
0x59	bString	(“NUL”)	0x00
0x5A	bString	ASCII character.	0x2D (“-“)
0x5B	bString	(“NUL”)	0x00
0x5C	bString	ASCII character.	0x53 (“S”)
0x5D	bString	(“NUL”)	0x00
0x5E	bString	ASCII character.	0x79 (“y”)
0x5F	bString	(“NUL”)	0x00
0x60	bString	ASCII character.	0x73 (“s”)
0x61	bString	(“NUL”)	0x00
0x62	bString	ASCII character.	0x74 (“t”)
0x63	bString	(“NUL”)	0x00
0x64	bString	ASCII character.	0x65 (“e”)
0x65	bString	(“NUL”)	0x00
0x66	bString	ASCII character.	0x6D (“m”)
0x67	bString	(“NUL”)	0x00
0x68	bString	ASCII character.	0x20 (“ “)
0x69	bString	(“NUL”)	0x00
0x6A	bString	ASCII character.	0x44 (“D”)
0x6B	bString	(“NUL”)	0x00
0x6C	bString	ASCII character.	0x65 (“e”)
0x6D	bString	(“NUL”)	0x00
0x6E	bString	ASCII character.	0x73 (“s”)
0x6F	bString	(“NUL”)	0x00
0x70	bString	ASCII character.	0x69 (“i”)
0x71	bString	(“NUL”)	0x00
0x72	bString	ASCII character.	0x67 (“g”)
0x73	bString	(“NUL”)	0x00
0x74	bString	ASCII character.	0x6E (“n”)
0x75	bString	(“NUL”)	0x00
USB String Descriptor - Index 2 (Product)			

Address	Field Name	Description	On-board Defaults
0x76	bLength	String descriptor length in bytes (restricted to 63 bytes or less).	0x28
0x77	bDescriptorType	Descriptor type	0x03
0x78	bString	ASCII character.	0x55 (“U”)
0x79	bString	(“NUL”)	0x00
0x7A	bString	ASCII character.	0x53 (“S”)
0x7B	bString	(“NUL”)	0x00
0x7C	bString	ASCII character.	0x42 (“B”)
0x7D	bString	(“NUL”)	0x00
0x7E	bString	ASCII character.	0x20 (“ “)
0x7F	bString	(“NUL”)	0x00
0x80	bString	ASCII character.	0x53 (“S”)
0x81	bString	(“NUL”)	0x00
0x82	bString	ASCII character.	0x74 (“t”)
0x83	bString	(“NUL”)	0x00
0x84	bString	ASCII character.	0x6F (“o”)
0x85	bString	(“NUL”)	0x00
0x86	bString	ASCII character.	0x72 (“r”)
0x87	bString	(“NUL”)	0x00
0x88	bString	ASCII character.	0x61 (“a”)
0x89	bString	(“NUL”)	0x00
0x8A	bString	ASCII character.	0x67 (“g”)
0x8B	bString	(“NUL”)	0x00
0x8C	bString	ASCII character.	0x65 (“e”)
0x8D	bString	(“NUL”)	0x00
0x8E	bString	ASCII character.	0x20 (“ “)
0x8F	bString	(“NUL”)	0x00
0x90	bString	ASCII character.	0x41 (“A”)
0x91	bString	(“NUL”)	0x00
0x92	bString	ASCII character.	0x64 (“d”)
0x93	bString	(“NUL”)	0x00
0x94	bString	ASCII character.	0x61 (“a”)
0x95	bString	(“NUL”)	0x00
0x96	bString	ASCII character.	0x70 (“p”)
0x97	bString	(“NUL”)	0x00
0x98	bString	ASCII character.	0x74 (“t”)
0x99	bString	(“NUL”)	0x00
0x9A	bString	ASCII character.	0x65 (“e”)
0x9B	bString	(“NUL”)	0x00
0x9C	bString	ASCII character.	0x72 (“r”)
0x9D	bString	(“NUL”)	0x00
0x9E-0x135	Not used		0xXX
Hardware Revision String Descriptor - Index 6			
0x136	bLength	String descriptor length in bytes (Hardware revision string).	0x0A
0x137	bDescriptorType	Descriptor type	0x03
0x138	bString		0x30
0x139	bString		0x00
0x13A	bString		0x30
0x13B	bString		0x00
0x13C	bString		0x30

Address	Field Name	Description	On-board Defaults
0x13D	bString		0x00
0x13E	bString		* 0x32 ** 0x33
0x13F	bString		0x00
* - First Silicon value ** - Second Silicon value			

Table 11 – String Descriptors

ATA/ATAPI Interface

The ATA/ATAPI port on the ISD-200 is compliant with the *Information Technology – AT Attachment with Packet Interface – 4 (ATA/ATAPI-4) Specification, T13/1153D Rev 18 (ftp://fission.dt.wdc.com/x3t13/project/d1153r18.pdf)*. The ISD-200 provides support for Packet commands as well as ATA commands. The ISD-200 provides a vendor-specific Identify (FBh) command that returns configuration data and USB descriptor information (see **Descriptors** section, page 7) from an attached mass storage device. Additionally, there is a built in vendor-specific event notify command to communicate certain events on an interrupt basis to the device.

Protocol

The ISD-200 supports command protocol flows as defined in the *ATA/ATAPI-4 Specification*. Commands are grouped into different classes, based on the protocol followed for command execution.

The ATA/ATAPI interface supports the following clarifications:

- Immediately after the reset recovery period, the ISD-200 will write 0x00 to the Device Control register.
- Arbitrary byte count transfers supported.
- 16-bit data reads and writes. 8-bit data transfers not supported.

Reset Mapping

The ATA/ATAPI Interface responds to several resets, Power-on, Resume, USB, MSC Hard, and Vendor-specific Soft reset.

In the case of a Power-on reset, a full device initialization is performed (See Figure 3). FBh data is retrieved and stored if applicable. In the case of a Resume reset, a full device initialization is performed as well (previously stored FBh data is not effected).

In the cases of USB reset and MSC Hard reset, a partial initialization is performed which excludes all attempts to perform Identify Device commands. If BUS_POWER=1 then the USB reset causes a full initialization after the USB configuration setting is restored.

In the case of a Vendor-specific Soft reset, only the internal state machines are reset.

Device Requirements

Attached mass storage devices must support the following device requirements.

ATA Power Supplied from the USB Bus
Power requirements for the device vary depending upon the mode used. If no serial ROM is present and IMODE is asserted, the device must be capable of operating at 4.4V when in enumeration mode (100 mA). That is, when obtaining configuration and USB descriptor information from the device using the vendor-specific Identify (FBh) command. After enumeration, the device must be able to operate at 4.75V when in operational mode (500 mA).

If a serial ROM is used, the device must be able to operate at 4.75V when obtaining configuration and USB descriptor information.

ATA Reset, A1h, FBh
BSY and DRQ must be cleared by the device prior to the amount of time specified by ATA Initialization Timeout configuration field has passed since the removal of reset.

ATA Polling Device
The device shall be capable of being a polling only device. As such, the ATA signal INTRQ is not required but may be useful as a debug tool.

ISD-200 ATA Bus State During Idle
The ISD-200 drives the address lines high and the last contents of the data register onto the ATA bus when idle. Pull up or pull down resistors shall not be used in bus power systems as the drive supply power will be shut off in USB suspend mode.

ATA PIO Support
Register accesses are performed in PIO mode 0 (750 ns cycle time). The maximum data register accesses are performed in PIO mode 2 (292 ns cycle time). The ISD-200 configuration bytes must be set such that data register accesses do not exceed the maximum speed supported by the device.

ATA Initialization Timeout

The ISD-200 supports a default configuration of 3.2 seconds for ATA Initialization Timeout. If a serial ROM is used, its ATA Initialization Timeout configuration value will override the default prior to device initialization. If no serial ROM is used and I_MODE is asserted, subsequent device initializations will use the ATA Initialization Timeout configuration value supplied by the vendor-specific Identify (FBh) command.

Reset Recovery shall be 3 ms. The following figure graphically defines “Initialization Timeout” and “Reset Recovery”.

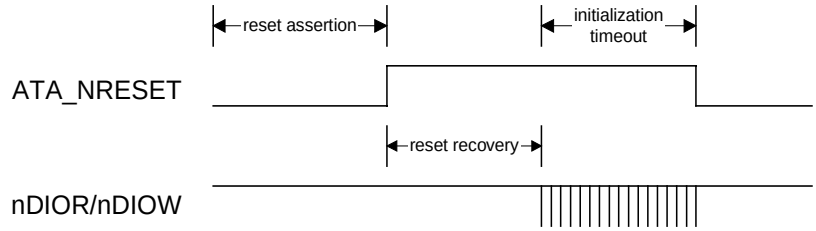


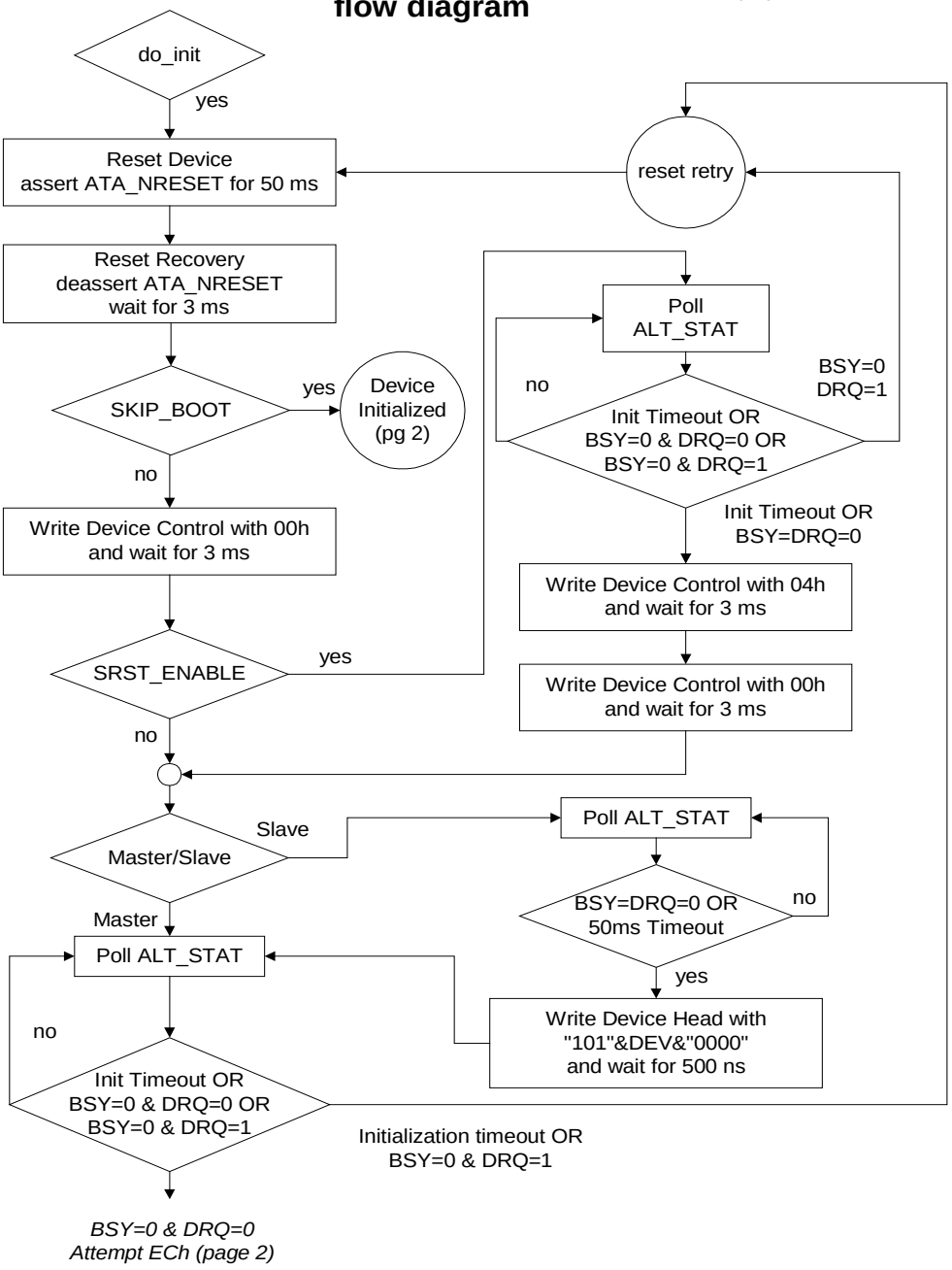
Figure 2 – ATA Reset Protocol

Device Initialization Sequence

The diagrams on the following pages show the normal sequence used for device initialization.

INITIALIZE flow diagram

page 1 of 2



INITIALIZE
flow diagram

page 2 of 2

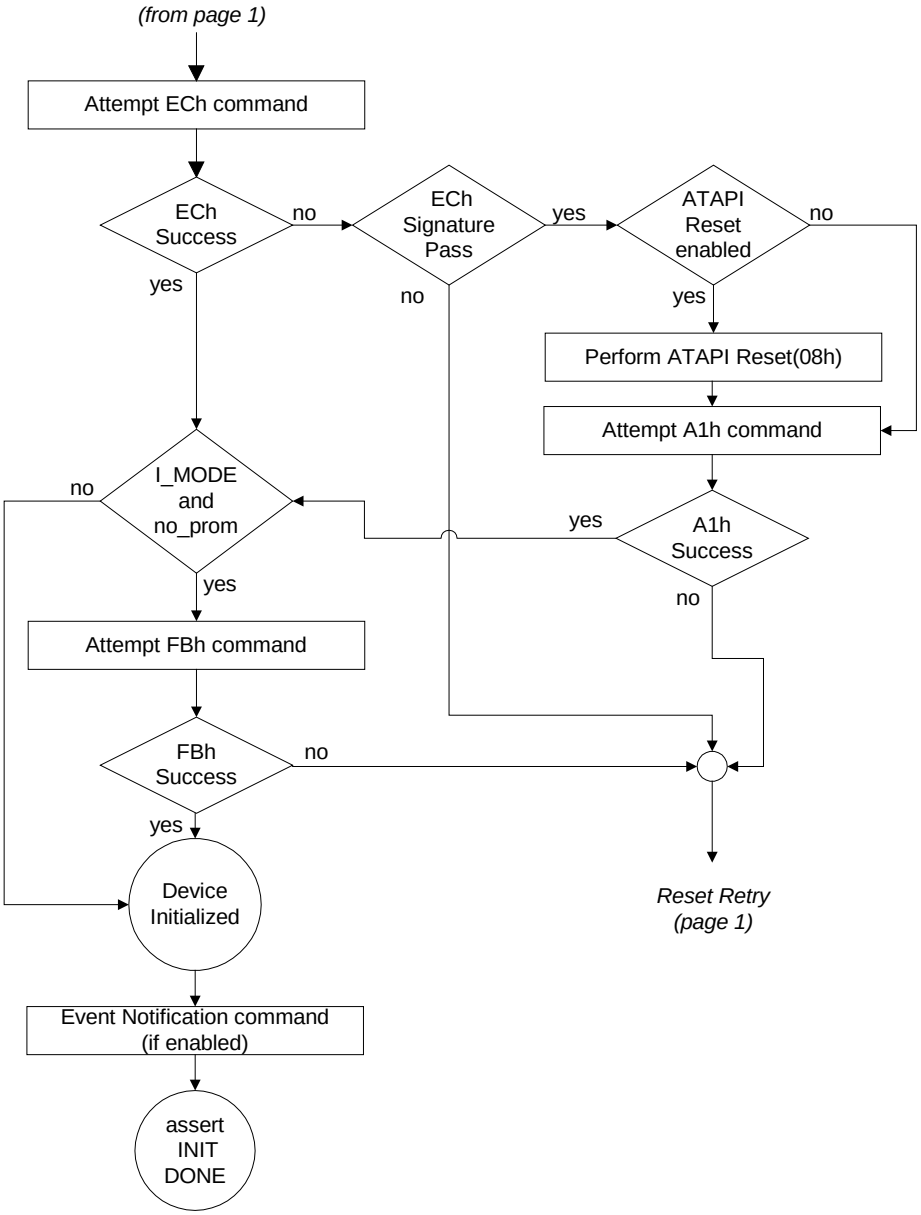


Figure 3 – Full Device Initialization Sequence

ATA Command Block

ATA commands for the ISD-200 shall be supported by command encoding in the command block portion of the *MSC* Command Block Wrapper *(CBW)*. Refer to the *USB Mass Storage Class (MSC) Bulk Only Transport Specification* for information on CBW formatting.

The ATA Command Block *(ATACB)* provides a means of passing ATA commands and ATA register accesses for execution. The *ATACB* resides in the *CBWCB* portion of the *CBW*. The *ATACB* shall be distinguished from other command blocks by the first two bytes of the command block matching the *wATACBSignature*. Only command blocks that have a valid *wATACBSignature* shall be interpreted as ATA Command Blocks. All other fields of the *CBW* and restrictions on the *CBWCB* shall remain as defined in the *USB Mass Storage Class Bulk Only Transport Specification*. The *ATACB* shall be 16 bytes in length. The following table and text defines the fields of the *ATACB*.

Byte	7	6	5	4	3	2	1	0
0-1	*wATACBSignature*							
2	Reserved (0)		*bmATACBActionSelect*					
3	*bmATACBRegisterSelect*							
4	*bATACBTransferBlockCount*							
5-12	*bATACBTaskFileWriteData*							
13-15	Reserved (0)							

Table 12 – ATA Command Block Formatting

Field Descriptions

wATACBSignature –

This signature indicates the *CBWCB* contains an *ATACB*. The signature field shall contain the value 2424h to indicate an *ATACB*. Devices capable of accepting only ATA Command Blocks shall return a command failed status if the *wATACBSignature* is not correct.

bmATACBActionSelect –

The bit fields of this register shall control the execution of the *ATACB*. Refer to the *ATACB* Command Flow diagram in section 4 of this document for further clarification . The bitmap of the *bmATACBActionSelect* shall be defined as follows:

Bits 7-6 Reserved - The host shall set these bits to zero.

Bit 5 *DEVOverride* – Use the DEV value specified in the *ATACB*.
0 = The DEV bit value will be determined from ISD200 Configuration data (byte 5 bit 5)
1= Then DEV bit value will be determined from the *ATACB*(0xB bit 5).

Bits 4-3 *DPErrorOverride(1:0)* - Device and Phase Error Override. These bits shall not be set in conjunction with *bmATACBActionSelect TaskFileRead*. The order of precedence for error override shall be dependant on the amount of data left to transfer when the error is detected, as depicted in the *ATACB* Command Flow diagram.
00 = Data accesses are halted if a device or phase error is detected.
01 = Phase error conditions are not used to qualify the occurrence of data accesses.

10 = Device error conditions are not used to qualify the occurrence of data accesses.
11 = Neither device error or phase error conditions are used to qualify the occurrence of data accesses.

Bit 2 *PollAltStatOverride* - Poll ALTSTAT Override.
0 = The Alternate Status registered shall be polled until BSY=0 before proceeding with the *ATACB* operation.
1 = Execution of the *ATACB* shall proceed with the data transfer without polling the Alternate Status register until BSY=0.

Bit 1 *DeviceSelectionOverride* - Device Selection Override. This bit shall not be set in conjunction with *bmATACBActionSelect TaskFileRead*.
0 = Device selection shall be performed prior to command register write accesses.
1 = Device selection shall not be performed prior to command register write accesses.

Bit 0 *TaskFileRead* - Read and return the task file register data selected in *bmATACBRegisterSelect*. If *TaskFileRead* is set, the *dCBWDataTransferLength* field must be set to 8.
0 = Execute *ATACB* command and data transfer (if any).
1 = Only task file registers selected in *bmATACBRegisterSelect* shall be read. Task file registers not selected in *bmATACBRegisterSelect* shall not be accessed and 00h shall be returned for the unselected register data.

bmATACBRegisterSelect –

Setting the appropriate bit fields shall cause the task file read or write register access to occur. Task file read data shall always be 8 bytes in length. Unselected task file register data shall be returned as 00h. Task file register accesses shall occur in sequential order as shown (Bit 0 first, Bit 7 last). The *bmATACBRegisterSelect* bitmap shall be as defined below.

Bit 0	(3F6h)	Device Control / Alternate Status
Bit 1	(1F1h)	Features / Error
Bit 2	(1F2h)	Sector Count
Bit 3	(1F3h)	Sector Number
Bit 4	(1F4h)	Cylinder Low
Bit 5	(1F5h)	Cylinder High
Bit 6	(1F6h)	Device-Head (See *bmATACBActionSelect(5)*)
Bit 7	(1F7h)	Command / Status

bATACBTransferBlockCount –

This value shall denote the maximum requested block size in 512 byte blocks. This variable shall be set to the value last used for “Sectors per block” in the SET_MULTIPLE_MODE command. Valid values are 1, 2, 4, 8, 16, 32, 64, and 128. Command failed status shall be returned if an invalid value is detected in the *ATACB*. Non-multiple commands shall set this value to 1 (block size of 512 bytes).

bATACBTaskFileWriteData –

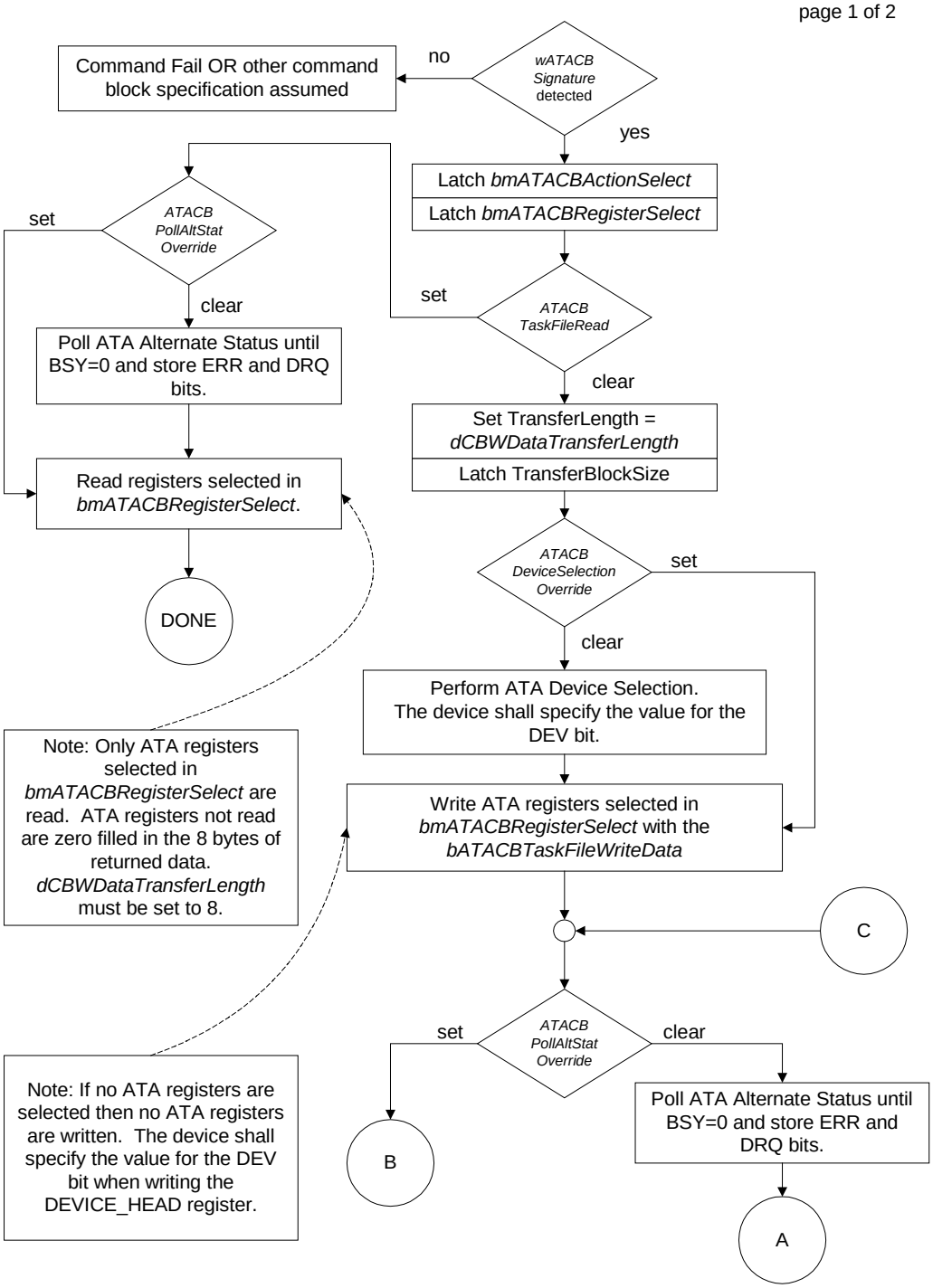
ATA register data used on ATA command or PIO write operations. Only data entries that have the associated *bmATACBRegisterSelect* bit set shall be required to have valid data.

ATACB Address offset 5h	(3F6h)	Device Control
ATACB Address offset 6h	(1F1h)	Features
ATACB Address offset 7h	(1F2h)	Sector Count
ATACB Address offset 8h	(1F3h)	Sector Number
ATACB Address offset 9h	(1F4h)	Cylinder Low
ATACB Address offset Ah	(1F5h)	Cylinder High
ATACB Address offset Bh	(1F6h)	Device-Head (See *bmATACBActionSelect(5)*)
ATACB Address offset Ch	(1F7h)	Command

ATA Command Flow

The following figure shows the flow of ATA commands, specifically the actions taken by the ISD-200 based upon how the ATA Command Block is configured.

page 1 of 2



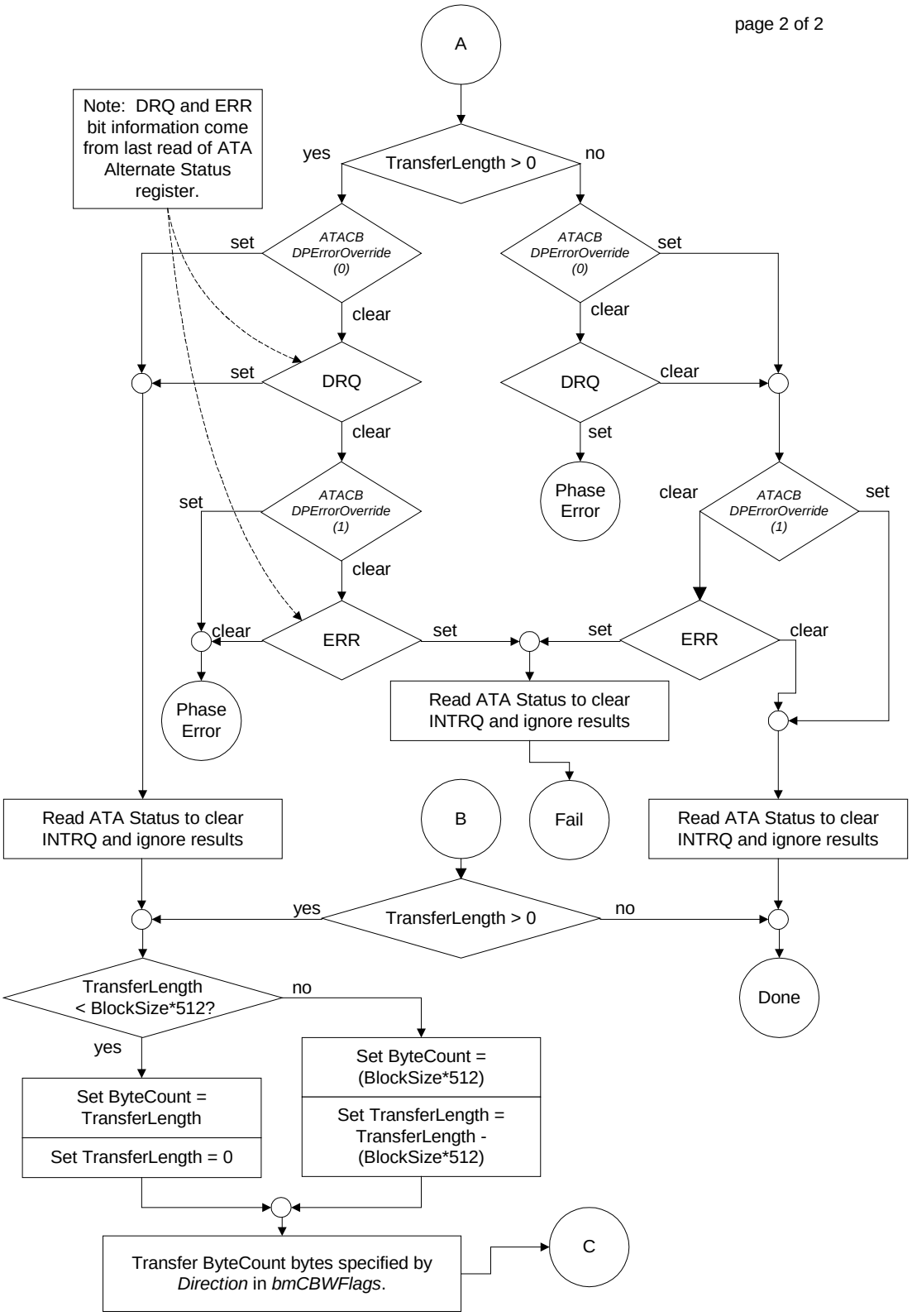


Figure 4 – ATA Command Block Flow Diagram

Vendor-Specific ATA Commands

There are two vendor-specific ATA commands implemented in the ISD-200. They are shown in the following table.

Label	Command Code	Description
IDENTIFY	FBh	This command is used to read ISD-200 configuration data and USB descriptor data from an attached mass storage device.
EVENT_NOTIFY	Specified in Configuration Data	This command communicates certain events to the device and is executed as the events occur.

Table 14 – Vendor-Specific ATA Commands

IDENTIFY
The vendor-specific Identify (FBh) command enables the ISD-200 to request configuration and USB descriptor information from an attached mass storage device.

Command Code
FBh

Feature Set
The PACKET Command feature set may or may not be implemented.

Protocol
PIO data-in (refer to *ATA/ATAPI-4 Specification*, section 9.7).

Input

Register	7	6	5	4	3	2	1	0
Features	N/A							
Sector Count	N/A							
Sector Number	N/A							
Cylinder Low	N/A							
Cylinder High	N/A							
Device/Head	obs	N/A	obs	DEV	N/A	N/A	N/A	N/A
Command	FBh							

Device/Head register –
The DEV bit indicates the selected device.

Normal Outputs

Register	7	6	5	4	3	2	1	0
Error	N/A							
Sector Count	N/A							
Sector Number	N/A							
Cylinder Low	N/A							
Cylinder High	N/A							
Device/Head	obs	N/A	obs	DEV	N/A	N/A	N/A	N/A
Status	BSY	N/A	N/A	N/A	DRQ	N/A	N/A	ERR

Device/Head register –
The DEV bit indicates the selected device.

Status register –
BSY shall be cleared to zero upon command completion.
DRDY shall be set to one.
DF (Device Fault) shall be cleared to zero.
DRQ shall be cleared to zero.
ERR shall be cleared to zero.

Error Outputs
If the device does not support this command, the device shall return command aborted. Otherwise, the device shall not report an error.

Description
When the command is issued, the device sets the BSY bit to one, and prepares to transfer 320 bytes of configuration/descriptor data to the ISD-200. The device then sets DRQ to one and clears BSY to zero.

The arrangement and meaning of the FBh data bytes are specified in Tables 3, 7-11. An example of FBh programming is shown in **Appendix A**.

EVENT_NOTIFY
The vendor-specific Event-notify command enables the ISD-200 to communicate the occurrence of certain events to the attached device. (See **Operational Modes, NEJECT, NCART_DET Pins**)

Command Code
Specified in the ISD-200 Configuration Bytes, address 0x2. Programming the command code to 0x00 disables the Event-notify feature.

Feature Set
The PACKET Command feature set is used.

Protocol
Non-data (refer to *ATA/ATAPI-4 Specification*, section 9.9).

Input

Register	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Features	USB Reset	Class Specific Reset	USB Suspend	USB Resume	Cartridge Insert	Cartridge Release	Eject Button Press	Eject Button Release
Sector Count	N/A							
Sector Number	N/A							
Cylinder Low	STATE0							
Cylinder High	STATE1							
Device/Head	N/A							
Command	Specified in the ISD-200 Configuration Bytes							

Features register –
The USB Reset bit indicates that a USB Reset event has occurred.
The Class Specific Reset bit indicates that an *MSC* Reset was issued by the host.
The USB Suspend bit indicates that the USB Bus has gone into suspend.
The USB Resume bit denotes that the USB Bus is no longer in suspend.
The Cartridge Insert bit is set when the device media is inserted.

The Cartridge Release bit is set when the device media is ejected.
The Eject Button Press bit is set when the eject button on the device is pressed.
The Eject Button Release bit is set when the eject button on the device is released.

Cylinder High –
The STATE0 vendor-specific field is combined with STATE1 specify state information to the attached device.

Cylinder Low –
The STATE1 vendor-specific field is combined with STATE0 specify state information to the attached device.

Normal Outputs

Register	7	6	5	4	3	2	1	0
Error	N/A							
Sector Count	N/A							
Sector Number	N/A							
Cylinder Low	NSTATE0							
Cylinder High	NSTATE1							
Device/Head	N/A							
Status	BSY	N/A	N/A	N/A	DRQ	N/A	N/A	N/A

Cylinder High –
The NSTATE0 vendor-specific field is combined with NSTATE1 to communicate information about the state of the device back to the host.

Cylinder Low –
The NSTATE1 vendor-specific field is combined with NSTATE0 to communicate information about the state of the device back to the host.

Status register –
BSY shall be cleared to zero upon command completion.
DRQ shall be cleared to zero.

Error Outputs
If the device does not support this command, the device shall return command aborted. Otherwise, the device shall not report an error.

Description
When this command is issued, the ISD-200 will wait until the device clears BSY and DRQ to zero before beginning the input register writes. After writing the input registers, the ISD-200 waits for BSY and DRQ cleared to zero and then reads the state information (NSTATE).

The event notification command is issued following every assertion of ATA_NRESET to the device and following the POR device initialization sequence. The event notification command is also issued after any of the events reported in the event notification data take place.

If any combination of mating events (mating events are defined as suspend/resume, cartridge insert/release, eject press/release, and USB/class reset) take place before the ISD-200 can issue the event notification command to the device, the following will occur:

1. Send an event notification command showing all events.
2. Send a subsequent event notification command showing only the most recent of any mated events.

If an event notification command does not complete for any reason (such as an incoming reset), the ISD-200 will re-issue the command (with any new event data) until it completes successfully. Success of the command does NOT depend upon the ERR bit. If the DRQ bit is set in response to an event notification, ISD-200 will continue to poll (in order to make device incompatibility obvious).

Power Management

BUS_POWER Pin

First silicon does not support the BUS_POWER pin being set (See **Errata #6**). The ISD-200 reacts to the following conditions as stated:

- When a USB suspend condition exists and BUS_POWER pin is asserted, the external clock source(CLKN) is disabled, and if remote wakeup is disabled the oscillator is turned off, all ATA outputs put into low power mode, and ATA control lines (nATA_RESET, nDIOR, nDIOW) are 3-stated to hi-Z.
- When a USB suspend condition exists the remainder of the ATA lines (nCS(1:0), DA(2:0), DD(15:0)) are driven low. First silicon **Errata #6** includes all ATA lines being three-stated only when BUS_POWER is asserted, and a suspend condition exists.
- When the BUS_POWER pin is asserted, the USB Configuration value is set to zero, and configuration byte 8 bit 4 is set, all ATA outputs are put into low power mode, ATA control lines (nATA_RESET, nDIOR, nDIOW) are 3-stated to hi-Z.
- When the USB Configuration value is set to zero, and configuration byte 8 bit 4 is set the remainder of the ATA lines (nCS(1:0), DA(2:0), DD(15:0)) are driven low. First silicon does not support this feature (See **Errata #6**).
- The ISD-200 resets itself in the following cases:
  - If BUS_POWER is asserted USB configuration is set to 2.
  - If BUS_POWER changes when the USB configuration is set to 0.
- The ISD-200 drives USB_ENUM before attempting to talk to attached device.
- The ISD-200 does not attempt to talk to the disk until the USB configuration is set to either 1 or 2. First silicon does not support this feature (See **Errata #6**).
- Setting the Descriptor Override bit is incompatible with setting BUS_POWER pin.

The assertion of BUS_POWER will cause a Get Status USB request from the host to report a '0' (indicating bus-powered status) in bit 0 of the information returned. The de-assertion of BUS_POWER will cause a Get Status USB request from the host to report a '1' (indicating self-powered status) in bit 0 of the information returned.

BUS_POWER pin state may also effect how USB descriptors are presented to the host (See **Selectable Device Power Configurations** section below). (See **Errata #6**)

NPWR500, and NPWR100 Pins

The NPWR500 and NPWR100 output pins indicate that the USB host has allotted the requested power(bMaxPower) to the peripheral system (See **Figure 5**). These pins allow USB power to be controlled in order to meet USB power consumption requirements. In the case of a USB Suspend condition, both NPWR500 and NPWR100 are de-asserted. Also, the ISD-200 will put itself into a low power state as described above. Upon a resume condition, the ISD-200 will resume normal operation and restore the NPWR500 and NPWR100 pin values as before suspend. ISD-200's power source is not to be controlled at any time using the NPWR500 or NPWR100 pins.

NPWR100 is always asserted unless in USB suspend. NPWR100 asserted indicates that up to 100 mA may be drawn from USB. NPWR500 is only asserted when the USB configuration is set to 1.

NPWR500 asserted indicated that up to 500 mA may be drawn from USB depending on what the bMaxPower value is set to for configuration 1

Selectable Device Power Configurations

The ISD-200 is capable of offering three types of system power configurations.

- Self-Powered – Current is not taken from USB
- Bus-Powered – Current is taken from USB
- Dynamically-powered – Current may be taken from USB
  The ISD-200 may be configured to operate from different power sources dynamically depending upon the presence of external (NOT USB) power. Power source configuration is accomplished by controlling the BUS_POWER input pin accompanied by the appropriate ISD-200 configuration and USB Descriptor settings. NPWR100 and NPWR500 are also used to control power consumption as described in the previous sub-section.

Self-Powered Example

The BUS_POWER pin is de-asserted.
ISD-200 Configuration:
Configuration 2 Present = 0
Report One Configuration = 0
Device Descriptor:
bNumConfigurations = 1
Configuration Descriptor:
bConfigurationValue1 = 1
iConfiguration1 = 3
bMaxPower1 = Required USB current limited to under 100 mA (May need to use NPWR500 to satisfy USB suspend current requirements if any current is drawn)
iInterface1 = 4
bConfigurationValue2 = N/A
iConfiguration2 = N/A
bMaxPower2 = N/A
iInterface2 = N/A

Bus-Powered Example

The BUS_POWER pin is asserted. (See **Errata #6**)
ISD-200 Configuration:
Configuration 2 Present = 0
Report One Configuration = 0
Device Descriptor:
bNumConfigurations = 1
Configuration Descriptor:
bConfigurationValue1 = 1
iConfiguration1 = 3
bMaxPower1 = Required USB current limited to under 500 mA (May need to use NPWR500 to satisfy USB suspend and pre-configuration current requirements)
iInterface1 = 4
bConfigurationValue2 = N/A
iConfiguration2 = N/A
bMaxPower2 = N/A
iInterface2 = N/A

Dynamically-Powered Example

The BUS_POWER pin is asserted to indicate the absence of external power and de-asserted to indicate the presence of external power. (See **Errata #6**)
ISD-200 Configuration (See **Table 3**):
'Configuration 2 Present' = 1
'Report One Configuration' = Pick either 0 or 1 depending on host software and desired results (See **Tables 3 and 8**).

Device Descriptor (See **Table 7**):

bNumConfigurations = 1 if 'Report One Configuration' is set otherwise 2

Configuration Descriptor (See **Table 8**):

bConfigurationValue1 = 1
iConfiguration1 = 3
bMaxPower1 = (USB Current required by the peripheral system limited to under 500mA)
iInterface1 = 4
bConfigurationValue2 = 2
iConfiguration2 = 7
bMaxPower2 = 0x31 (98 mA)
iInterface2 = 8

Configuration 1 = Dynamic Configuration (NPWR500 asserted)

Configuration 2 = Self-Powered Configuration

Configuration 2 consists of configuration 1 values except with the 4 bytes of override values from bConfigurationValue2, iConfiguration2, bMaxPower2, and iInterface2 (See **Tables 3 and 7-8**).

If 'Report One Configuration' is set to 1:

- Configuration 1 will be returned when BUS_POWER is asserted.
- Configuration 2 will be returned when BUS_POWER is de-asserted.

If 'Report One Configuration' is set to 0:

- When BUS_POWER is asserted, bNumConfigurations in the device descriptor will be overridden with 1 and Configuration 1 will be returned.
- When BUS_POWER is not asserted Configuration 1 and Configuration 2 will be returned.

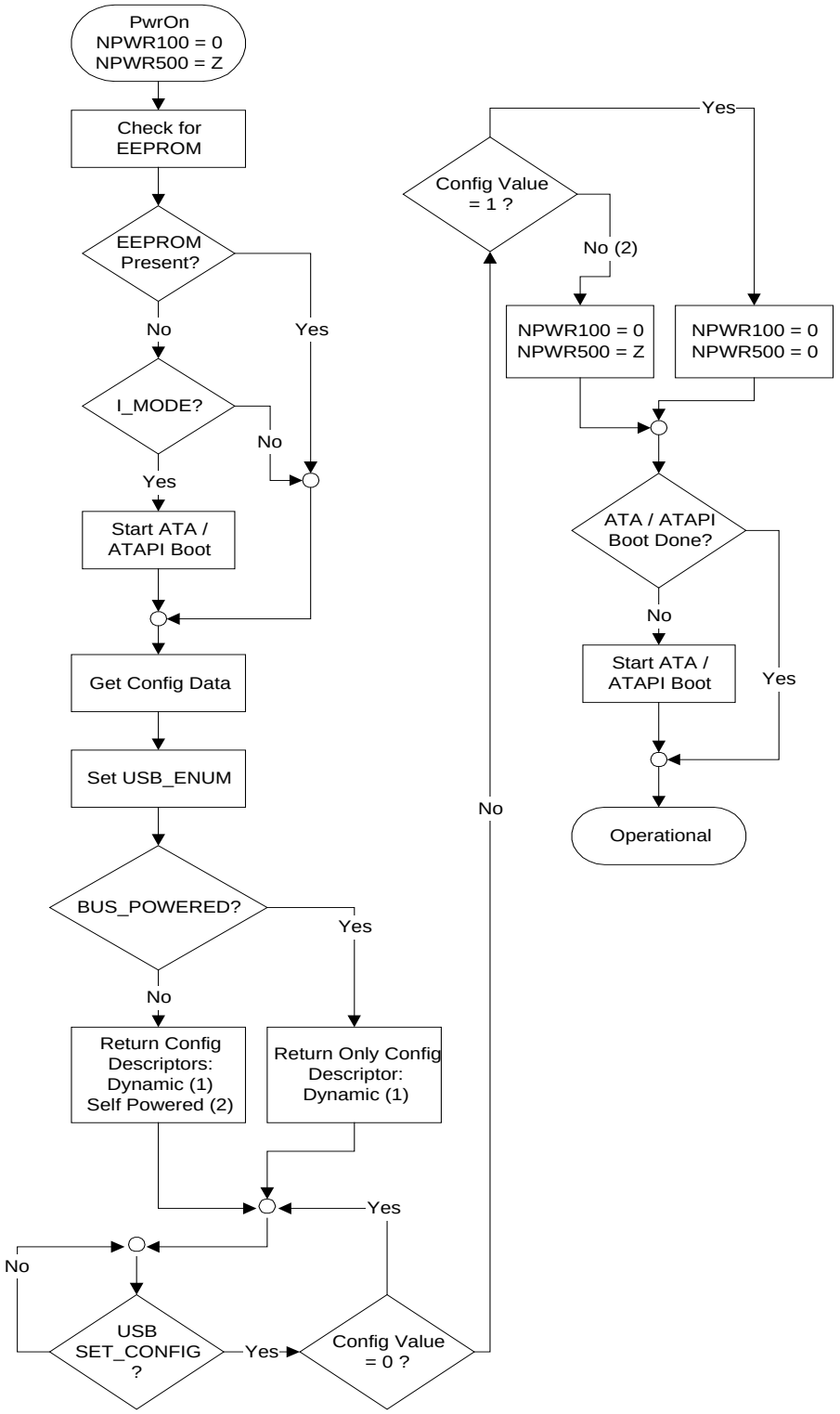


Figure 5 – Configuration Sequence

Operational Modes

NEJECT & NCART_DET Pins

These pins are used to trigger remote-wakeup (See **Table 8, 0x2F**) as well as Event Notification (See **Vendor Specific ATA Commands, EVENT_NOTIFICATION**). When asserted low 'NEJECT' indicates to ISD200 that an eject button has been pushed. When asserted low, 'NCART_DET' indicates that a cartridge is present. There is an internal 1ms filter on each of these inputs.

I_MODE Pin

I_Mode pin, when asserted high, allows the ISD200 Configuration and USB Descriptor data to be retrieved from an attached device. (See **ISD-200 CONFIGURATION, DATA SOURCES**)

ATA_EN Pin

ATA_EN pin allows ATA bus sharing with other host devices. De-asserting (ATA_EN=0) causes the ISD-200 to 3-state all ATA bus interface pins to hi-Z; de-assert USB_ENUM, and reset all logic *except on-board ROM / serial ROM logic that loads configuration data.* This logic remains enabled to allow configuration data loads for the configurable external clock (CLKN) upon occurrence of chip reset. Asserting ATA_EN (ATA_EN=1) allows normal operation. In order to insure the internal pull-up for ATA_EN is on, TEST(3) must be tied low.

Test Mode Pins

TEST(3:0)	Mode Description
0000	**Normal Mode**. This is the default mode of operation, or run time mode. Pull-downs are on. CLKN is disabled.
0001	**Normal Mode**. This is the default mode of operation, or run time mode. Pull-downs are on. CLKN defaults to 32 MHz.
0010	**Normal Mode**. This is the default mode of operation, or run time mode. Pull-downs are on. CLKN defaults to 40 Mhz.
0011	**NandTree** – Allows board level manufacturing tests. See following section.
0100	**Scan** M**ode –** Fab only test mode
0101	**Limbo** Setting this mode disables all output (3-state to hi-Z)
0110	**Disable Disk (Normal) -** Enable USB enumeration without ATAPI interface. Pull-downs are on.
0111	**Reserved**
1000	**InTest (T)** - Functional test mode with shortened timers. The DPLL and OSC circuits are still powered with CLKN disabled. Pull-downs are off.
1001	**InTest (TM)** - Functional test mode with shortened timers and shortened RAM. The DPLL and OSC circuits are still powered with CLKN defaulting to 32 MHz. Pull-downs are off.
1010	**InTest (TMS)** - Functional test mode with shortened timers, shortened RAM, and skip ATAPI identify boot sequence. The PLL and OSC circuits are still powered with CLKN defaulting to 40 MHz. Pull-downs are off.
1011	**TestMux –** Fab only test mode
1100	**Scan** M**ode –** Fab only test mode
1101	**InTest (BT)** – Functional test mode with DPLL bypassed (m=n=1), OSC circuits disabled (pass through), and shortened timers. Pull-downs are off.
1110	**Reserved**

1111	**PowerDown –** Fab only test mode

Table 13 – Test Modes

Test Pin Pull-downs
In order to insure the internal pull-downs for TEST(2:0) are on, TEST(3) must be tied low.

Normal Mode With CLKN Enabled
There are two modes in which the CLKN output clock is initiated before ISD-200 configuration data is received. The main reason for providing a clock in this manner is to allow an ATA device to use this clock as a system clock when the ISD-200 configuration data source is to come from the device (See the **Vendor-Specific ATA Commands, Identify** section). Two clock frequencies are provided 32 and 40 MHz.

Disable Disk Mode
This mode allows the ISD-200 to temporarily bypass the normal device initialization in a manufacturing environment in order to program the EEPROM over USB. This mode is NOT to be used as a normal functional mode.

NandTree Test Mode
This mode disables all outputs except 'USB_ENUM' (NandTree output), allowing for testing of input connectivity. The list below shows the connectivity order of the NandTree chain (beginning to end).

NRESET,
ATA_EN,
IORDY,
DD[15:0], Note: DD[0] first, DD[15] last
NEJECT,
NCART_DET,
BUS_POWER,
SCAN_EN,
I_MODE,
SDA

Input pin connectivity can be tested with the following procedure:

1) Set all inputs on the chain to '1'. Output will be '1'.
2) Set NRESET to '0'. Output will toggle
3) Set NRESET back to '1'. Output will toggle.
4) Set '0' on the NandTree chain inputs from the beginning of the chain to the end (in order). The output will toggle with each input toggle, testing pad / IO cell connectivity.

Limbo Mode
ISD-200 provides a "limbo mode" in which all of its output pads are placed in a high-impedance state.

External Circuitry

External Components Connection

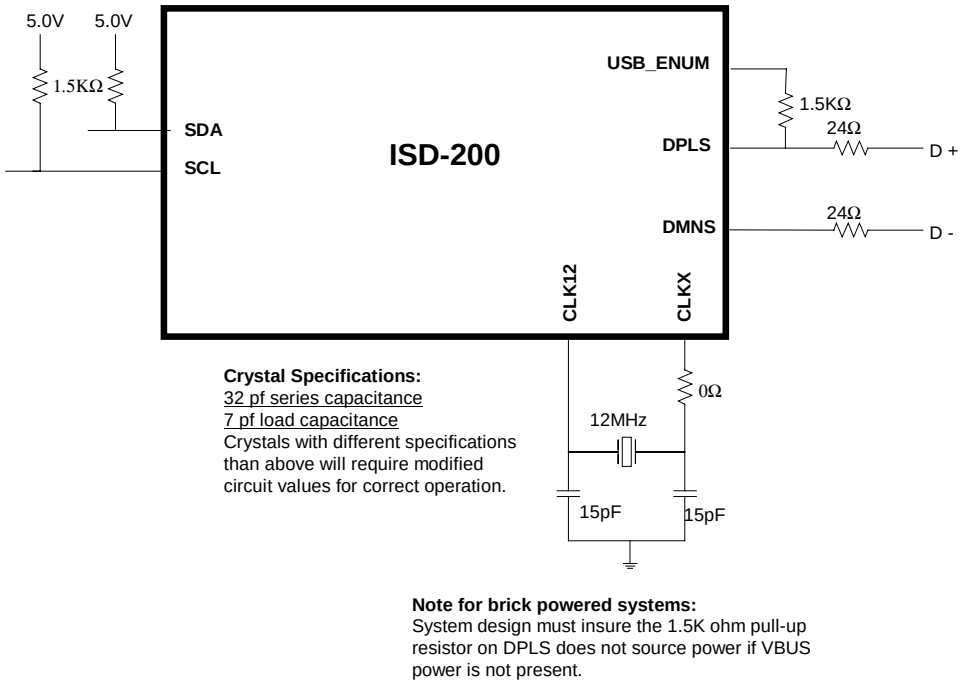


Figure 6 – External Components Connection

IORDY Hookup

The IORDY pin must be pulled up using a 1 K ohm resistor. The figure on the left is used when the ISD-200 and another device (or devices) is connected to the ATA bus of the drive. The figure on the right is used when the ISD-200 is the only device connected.

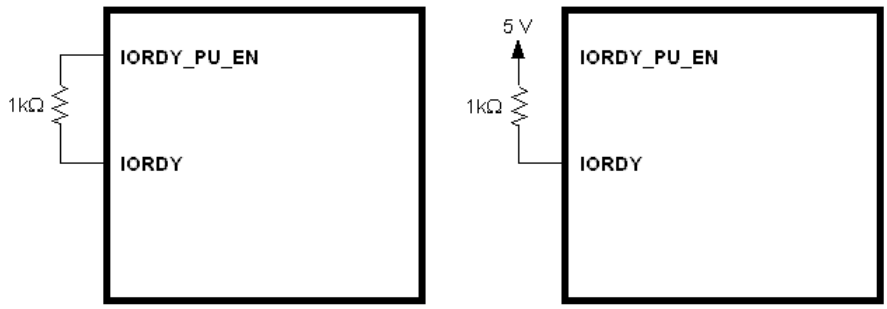


Figure 7 – IORDY Hookup

PDIAG and PDASP Hookup

Power on diagnostics timeout (750 ms and 30 s) must be limited to avoid an ISD-200 timeout. One method to accomplish this is to make sure PDIAG and PDASP are grounded on the circuit board and the device is wired to be master.

Absolute Maximum Ratings

Stresses in excess of the absolute maximum ratings can cause permanent damage to the device. These are absolute stress ratings only. Functional operation of the device is not implied at these or any other conditions in excess of those given in the operations sections of this data sheet. Exposure to absolute maximum ratings for extended periods can adversely affect device reliability.

Symbol	Parameter	Min	Max	Units
VDD33	3.3 V IO Supply	-0.5	4.0	Volts
VDD18	1.8 V IO Supply	-0.5	2.0	Volts
Vin	5V Tolerant Input Pin Voltage 3.3V Input Pin Voltage	-0.5 -0.5	5.5 V_{DD33} + 0.5	Volts
Iin	Input Pin Current	-20	20	mA
Ta	Ambient Operating Temperature Range	0°	70°	Celsius
Tstrg	Storage Temperature	-65	150	Celsius

Table 15 – Absolute Maximum Ratings

Electrical Characteristics

Voltage Parameter	Symbol	Test Conditions	Min	Typ	Max	Unit
Input Voltage Low	V_{IL}	—	—	—	0.8	V
Input Voltage High	V_{IH}	—	2.0	—	—	V
Output Voltage Low	V_{OL}	—	—	—	0.4	V
Output Voltage High	V_{OH}	—	2.4	—	—	V
Power Supply Voltage	V_{DD33}	—	3.0	3.3	3.6	V
	V_{DD18}	—	1.65	1.8	1.95	V

Note: (T_A = 0 °C, V_{DD33} = 3.3 V ± 0.3 V, V_{SS} = 0 V)

Table 16 – DC Characteristics

Current Parameter	Operating	Suspend
V_{DD33} supply current (I_{DD33})	10 mA (typ)	6 µA (typ)
V_{DD18} supply current (I_{DD18})	10 mA (typ)	8 µA / 275 µA (typ)
I_{DD33} with ATA_EN = 0	6 µA (typ)	N/A
I_{DD18} with ATA_EN = 0	1 mA (typ)	N/A

Note: (T_A = 0 °C, V_{DD} = 3.3 V ± 0.3 V, V_{SS} = 0 V), (V_{DD18} suspend current: BUSPWR=1, with out / with USB remote wakeup enabled in device)

Table 17 – Power Supply Current

Timing Characteristics

I²C Memory Device Interface Timing

The I²C memory device interface supports the I²C "fast mode." Timing specifics are given below.

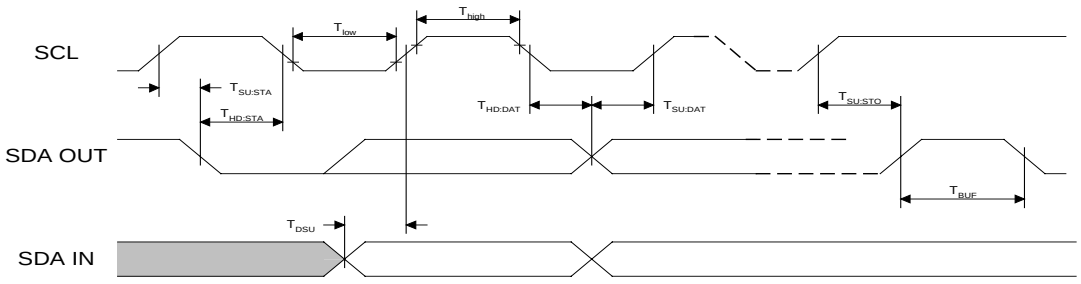


Figure 8 – I²C Memory Device Interface Timing

Parameter	Symbol	Value
Clock high time	T_{high}	667 ± 20 ns
Clock low time	T_{low}	1,333 ± 20 ns
Start condition hold time	$T_{HD:STA}$	667 ± 20 ns
Start condition setup time	$T_{SU:STA}$	667 ± 20 ns
Data output hold time	$T_{HD:DAT}$	667 ± 20 ns
Data output setup time	$T_{SU:DAT}$	667 ± 20 ns
Stop condition setup time	$T_{SU:STO}$	667 ± 20 ns
Required data valid before clock	T_{DSU}	84 ns
Bus free time	T_{BUF}	1,333 ± 20 ns

Table 18 – I²C Memory Device Interface Timing

USB Transceiver Timing Characteristics

The USS-725 USB transceiver complies with the timing and electrical requirements of the *Universal Serial Bus Specification version 1.0*.

ATA/ATAPI Port Timing Characteristics

All input signals on the ATA/ATAPI port are considered to be asynchronous, and are synchronized to the chip's internal system clock. All output signals are clocked using the chip's internal system clock, for which there is no external reference. Thus, the output signals should be considered asynchronous.

PIO mode 0 (750 ns cycle time) shall be used during power on reset (POR) and for non-data register accesses. Following POR, the PIO mode used for data register accesses is specified in the ISD-200 configuration bytes.

Clock

	Frequency	Duty Cycle
external crystal	12 MHz ± 0.25%	n/a

Note: Clock signal frequency is measured at $V_{DD18}/2$ point. Rise and fall times should be 2 ns or less.

Table 19 – Clock Requirements

Reset

The ISD-200 requires an off-chip power-on reset circuit. The supply voltage should be stable for a minimum of 1 ms prior to the release of nRESET.

Physical Diagrams

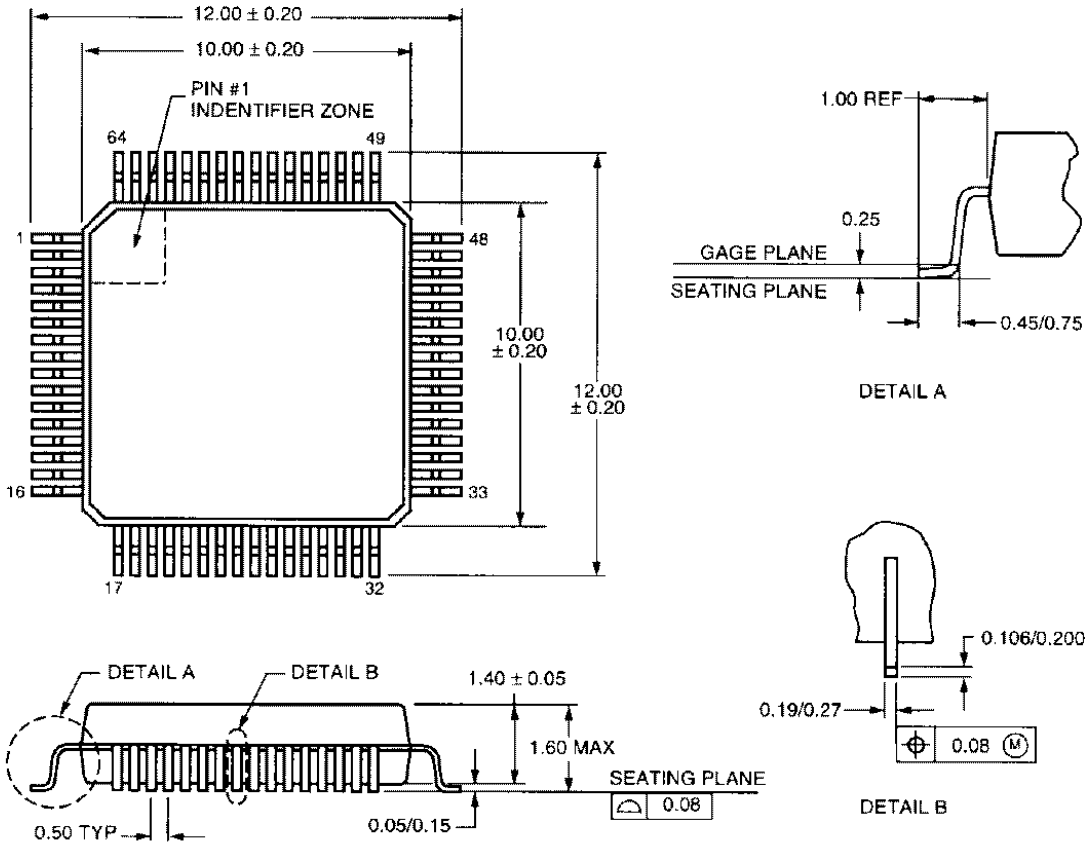


Figure 9 – Package Outline Diagram

Device Errata for First Silicon ISD-200 “0002”

This section identifies known problems with ISD-200 first silicon. Silicon revision information is obtained from a vendor specific USB descriptor request. ISD-200 first silicon returns the English Unicode string “0002” to the vendor specific request for string descriptor index 6. Second silicon returns “0003 for the same string. BCDDevice from internal ROM contents also indicates revision. First silicon returns \x0100 and second silicon returns \x0110. Follow on revisions of the ISD-200 will return different descriptor string values. Note the revision string may also be obtained by reading and decoding addresses 0x139-0x13F from internal ROM contents.

1. CLKN output may initialize with unknown frequency with ASIC in reset

The DPLL circuitry is not asynchronously reset, and when the TEST pins are set to generate CLKN output prior to the DPLL configuration data load, spurious frequencies may be generated from the CLKN output prior to the state machines loading the correct DPLL *m* and *n* parameters.

This anomaly only effects I_MODE operation. Setting the TEST pins for no CLKN generation and waiting for the configuration data load from EEPROM or on-board ROM eliminates the issue.

2. ATA_EN internal pull-up resistor is disabled during USB suspend operation

Internal logic incorrectly disables the internal pull-up resistor when the ISD-200 operates in USB suspend mode.

The issue can be addressed with an external pull-up resistor.

3. EEPROM write operation fails on last byte of last page with some EEPROM devices

The ISD-200 always increments the address when checking status of the previous write operation. On page boundaries, address bits 10:8 (or the chip select portion of the address with typical 256 byte page devices) are incremented prior to the status phase of the last byte write operation. Vendors of I2C devices that treat this address / status phase data as “don’t care” are compatible with the ISD-200. Vendors that do examine the address bits during the write completion check do not “ACK” the last write, as the address is now out of range for the device.

The issue is addressed by not writing the last byte in the last EPROM page. In almost all applications only a portion of the serial memory is utilized, and the problem does not surface. String descriptors can easily be modified to accommodate the loss of one byte (if need be).

4. SCL output does not 3-state to hi-Z when ‘1’, limiting SCL Vhi voltage to 3.3V

This issue may cause compatibility issues for certain EEPROM devices, as most data serial ROM data sheets report a Vhi voltage of 0.7*VCC requirement for correct operation. To operate at 400Khz (ISD-200 requirement), most devices require VCC set to 5V, which translates to a Vhi requirement of 0.7*5V=3.5V.

“Catalyst” EEPROM’s have been identified not to have the 0.7*VCC Vhi limitation. A field engineer states Catalyst’s input stage triggers Vhi at .5*VCC, or 2.5V. Regression testing with Catalyst parts found Zero failures. If another vendor is chosen, Vhi switching requirements must be investigated to insure proper operation.

5. Current draw of 5-6 mA when ATA_EN = 0

The ATA cells are not powered down when ATA_EN is inactive. When enabled, the ATA cell active circuitry typically draws between 5-6 mA of quiescent current (all cells combined). Disabling the active circuitry reduces quiescent current to approximately 4 uA (note the inputs must be tied high or low to keep them out of their high current switching region).

The TEST[3:0] inputs will place the ATA cells in low power mode if used in conjunction with ATA_EN. Setting TEST[3:0] = "1111" with ATA_EN = 0 will place the ATA cells in low power mode. Quiescent current will then be approximately 1 mA. This solution is compatible with rev B silicon (although not required).

6. USB Bus powered system support not possible with ISD-200 first silicon.

There are several issues that prevent the ISD-200 from supporting USB Bus powered systems. These issues only effect USB Bus powered operation. Brick powered system support is not effected.

ISD-200 rev A silicon can only be used in brick powered systems. The rev B version of the ISD-200 will address the issues and thus support USB Bus powered system integration.

7. USB Interrupt pipe support required for Mac OS drivers / applications

Although the USB Interrupt endpoint in the ISD-200 always returns 0x00's when addressed, some software / drivers used with Mac OS version expect the endpoint for proper operation. *This is not a HW issue with the ISD-200; it has been implemented as a work-around solution for legacy software issues.*

For legacy driver compatibility reasons, the interrupt endpoint should be reported in the ISD-200 USB descriptor information.

8. Support not provided for 16 byte ATAPI command block transfers

Previous revisions of this data sheet state the ISD-200 can support 16 byte ATAPI command blocks. The ATAPI Command Block Size configuration bits (configuration byte 0x05, bits[7:6]) must be set to "00", as only 12 byte ATAPI command block are supported by the ISD-200.

Advertisement of this feature has been removed from the data sheet. The ISD-200 will not support 16 byte ATAPI command block sizes. *No changes are planned for rev B silicon*.

9. CLKN does not get the correct frequency in all cases

See Errata #101

10. Failure to override descriptors for slave device when BSY=0 and DRQ=1.

See Errata #102.

11. Possible hang in ATACB functionality.

See Errata #103.

Device Errata for Second Silicon ISD-200 “0003”

100. nATA_Reset, nDIOR, and nDIOW allow parasitic current draw during suspend

When suspended second silicon attempts to reduce suspend current and allow flexibility by three-stating the three control lines nATA_Reset, nDIOR, and nDIOW. The cells leak about 2-3 uA that hold the cells from going into a low power state. The extra 3.3v supply current per cell is around 200 uA when in this high current consumption state. When there is a 1.5k pull-down in place, the cell draw much less current. If all cells have been placed in the low current consumption state then total 3.3v supply current should be around 200 nA. A weak pull-up could also be used to pull these signals up to reduce the 3.3v supply current to about 12 uA.

101. CLKN does not get the correct frequency in all cases

The following table indicates the source of CLKN configuration. “X” means CLKN does not run. Note: Columns indicate a sequence of actions from left to right. Rows indicate configuration source and BUS_POWER pin state.

	POR, ATA_EN=1	ATA_EN=0	ATA_EN=1	POR, ATA_EN=0	ATA_EN=1	ATA_EN=0
BUS_POWER=0, ATA HIGH POWER=0(byte 8, bit 4)						
E^2	EE CLKN	EE CLKN	EE CLKN	X	EE CLKN	EE CLKN
E^2 + Test pins	EE CLKN	EE CLKN	EE CLKN	Test CLKN	Test CLKN -> EE CLKN	EE CLKN
Test pins	Test CLKN	Test CLKN	Test CLKN	Test CLKN	Test CLKN	Test CLKN
I_MODE + Test pins	Test CLKN	Test CLKN	Test CLKN	FBh CLKN	FBh CLKN	FBh CLKN
BUS_POWER=1, ATA HIGH POWER=0(byte 8, bit 4)						
E^2	EE CLKN	X	EE CLKN	X	EE CLKN	X
E^2 + Test pins	EE CLKN	X	EE CLKN	X	EE CLKN	X
Test pins	Test CLKN	X	Test CLKN	X	Test CLKN	X
I_MODE + Test pins	Test CLKN	X	Test CLKN	X	FBh CLKN	X
BUS_POWER=0, ATA HIGH POWER=1(byte 8, bit 4)						
E^2	12MHz -> X	X	X	X	EE CLKN	EE CLKN
E^2 + Test pins	EE CLKN	EE CLKN	EE CLKN	Test CLKN	Test CLKN -> EE CLKN	EE CLKN
Test pins	Test CLKN	Test CLKN	Test CLKN	Test CLKN	Test CLKN	Test CLKN
I_MODE + Test pins	Test CLKN	Test CLKN	Test CLKN	FBh CLKN	FBh CLKN	FBh CLKN
BUS_POWER=1, ATA HIGH POWER=1(byte 8, bit 4)						
E^2	X	X	X	X	X	X
E^2 + Test pins	X	X	X	X	X	X
Test pins	Test CLKN	X	Test CLKN	X	Test CLKN	X
I_MODE + Test pins	Test CLKN	X	Test CLKN	X	FBh CLKN	X
Note: All values indicate CLKN output before USB Configuration is set. When ATA HIGH POWER=1 and BUS_POWER=1, the CLKN output will not turn on until the USB configuration is set to a non-zero value. Refer to the "BUS_POWER=1, ATA HIGH POWER=0" section in order to tell what the CLKN output will be when the USB Configuration is set.						

For devices using CLKN that allow ATA_EN to be deasserted, it is suggested that the BUS_POWER pin is deasserted in conjunction with ATA_EN if BUS_POWER is asserted itself.
If a device uses both test pins and E^2 or FBh to configure CLKN then it is suggested that both CLKN values match.
If a device is to use E^2 CLKN values then the test pins must be set to either CLKN test mode in order to get the expected value. If ATA_EN is allowed to deassert then there is the possibility that the test pin CLKN values will be used.

102. Failure to override descriptors for slave device when BSY=0 and DRQ=1.

This issue is regarding ISD200 Configuration byte 8 bit 3. This case will usually only occur when the device is a slave and there is a host side pull-down on DD7 and no pull-down on DD3. Is this case ISD200 will never enumerate.

Potential fixes:

1. Pull-up on DD7
2. Pull-down on DD7 and DD3

103. Possible hang in ATACB functionality.

If bATACBTransferBlockCount is set to zero, PollAltStatOverride is set to one and the CBW DataTransferLength not zero, the ATACB state machine will hang.

This should not be an issue if the bATACBTransferBlockCount is never set to zero or if there is never a data phase(CBW DataTransferLength=0) when PollAltStatOverride is set to one.

Appendix A – Example EEPROM or FBh Data Contents

Address	Field Name	Description	Example SROM / FBh Data
0x00	Data Signature (LSB)	This field specifies the least significant byte of the Serial ROM/FBh signature. **This register does not exist in HW (no POR values)**	0x52
0x01	Data Signature (MSB)	This field specifies the most significant byte of the Serial ROM/FBh signature. **This register does not exist in HW (no POR values)**	0x48
0x02	Event Notification	This field specifies the ATA event notification command. Setting this field to 0x00 disables this feature. **POR configuration default of 0x00**	0xFC
0x03	DPLL Parameters	This field denotes the parameters used by the internal DPLL. The original clock source is 12 MHz. 5 bits M (7:3), 2 bits N (2:1), 1 bit Enable (0). When enabled, multiply the original clock source by M, divided by N M: 00000 => M=1 00001 => M=1 00010 => M=2 00011 => M=3 …. 11111 => M=31 N: 00 => N=1 01 => N=3 10 => N=2 11 => N=4 Enable: 0 => CLK_N disabled 1 => CLK_N enabled **POR configuration default of 0x00**	0x43
0x04	ATA Initialization Timeout	This field specifies the time in multiples of 128 ms (0x19 = 3.2s) before the ISD-200 stops polling the Alternate Status device register for reset complete and restarts the reset process. NOTE: The ROM contents ATA Initialization Timeout value must be large enough to accommodate I_MODE operation during the first device initialization sequence (before FBh configuration data load) **POR configuration default of 0x02**	0x02
0x05	Reserved – Bits [7:6] Master/Slave Selection – Bit [5] ATAPI DEVICE RESET – Bit [4] ATA Timing – Bits [3:0]	Reserved; set to '0' This bit specifies device number selection. "0" - Drive 0 "1" - Drive 1 This bit specifies that the ISD-200 perform a ATAPI DEVICE RESET command during a full initialization sequence. This field determines ATA Bus data access cycle times. 0000 reserved 0001 Mode 2 (292 ns) 0010 Mode 2 (333 ns) 0011 Mode 2 (375 ns) 0100 Mode 1 (458 ns) 0101 Mode 1 (500 ns) 0110 Mode 1 (542 ns) 0111 Mode 1 (583 ns) 1000 Mode 0 (625 ns) 1001 Mode 0 (666 ns) 1010 Mode 0 (708 ns) 1011 Mode 0 (750 ns) 1100 reserved 1101 reserved 1110 reserved 1111 reserved **POR configuration default of 0x0B**	0x01
0x06	ATA Command Designator (Byte 0, LSB)	This field specifies the value in CBW CB field that designates if the CB is decoded as ATA commands instead of the ATAPI command block. **POR configuration default of 0x00**	0x24

Address	Field Name	Description	Example SROM / FBh Data
0x07	ATA Command Designator (Byte 1, MSB)	This field specifies the value in CBW CB field that designates if the CB is decoded as ATA commands instead of the ATAPI command block. **POR configuration default of 0x00**	0x24
0x08	Initialization Status – Bit[7]	This (Read Only) bit denotes the Device Initialization Status. If set, indicates the device initialization sequence is active.	0x40
	Configuration Descriptor 2 Present – Bit[6]	This bit specifies that a second Configuration Descriptor is present. See "BUS_POWER pin" under the "Power Management" section. "0" - not present "1" - present	
	Skip ATA / ATAPI Device Initialization – Bit[5]	This bit specifies that the ISD-200 skip device initialization. "0" - normal operation "1" - only reset the device prior to allowing USB enumeration. The function of this bit is overridden (0) if I_MODE is set. Notification command, descriptor override, SRST enable, ATAPI DEVICE RESET settings are irrelevant.	
	ATA HIGH POWER Device for USB Bus Powered Devices – Bit[4]	Setting this bit indicates that when USB Bus powered, the system's ATA interface is only powered when the NPWR500 pin is active (not controlled by NPWR100). "0" – normal operation "1" – If (BUS_POWER = 1) and (USB configuration = 0) then ATA pads are put into low power mode, ATA control lines (RESET-, DIOR-, DIOW-) are 3-stated to hi-Z, and the remainder of the ATA lines(NCS(1:0), DA(2:0), DD(15:0), and IORDY_PU_EN) are driven low. (See **Power Management**)	
	Descriptor Override – Bit[3]	Setting this bit causes the ISD-200 to override the USB Descriptors for ATA devices. Not compatible with setting BUS_POWER=1. If set, (I_MODE = 0), AND (ATA device identified or device initialization fails), bDeviceClass returned as 0xFF bDeviceSubClass returned as 0x00 bDeviceProtocol returned as 0xFF bInterfaceClass returned as 0xFF bInterfaceSubClass returned as 0x00 bInterfaceProtocol returned as 0xFF idProduct bit 0 is replaced with a '1' (Suggested: even idProduct for ATAPI devices, odd numbers for ATA devices) else USB descriptor information unaltered	
	Last LUN Identifier – Bits[2:0]	This field denotes the (zero-based) maximum number of LUNs supported. **POR configuration default of 0x30**	
0x09	Reserved – Bits[7:2]	Reserved; set to '0'	0x02
	Report One Configuration – Bit[1]	This bit determines whether one or two configuration descriptors are made available to the host. This bit should not be set if the Configuration Descriptor 2 Present configuration bit is not set. "0" - Both descriptors are reported to the host if BUS_POWER=0 "1" - Report only the configuration descriptor that corresponds to state of the BUS_POWER input. Configuration 1 is returned when BUS_POWER=1, configuration 2 is returned when BUS_POWER=0	
	SRST Enable – Bit[0]	Setting this bit enables the SRST reset algorithm in the ISD-200. **POR configuration default of 0x00**	
0x0A	Language ID String (0)	Address location for LANGID string (divided by 2). This index must be set to 0x00 if the string is not present.	0x28
0x0B	Manufacturer String (1)	Address location for iManufacturer string (divided by 2). This index must be set to 0x00 if the string is not present.	0x2A

Address	Field Name	Description	Example SROM / FBh Data
0x0C	Product String (2)	Address location for iProduct string (divided by 2). This index must be set to 0x00 if the string is not present.	0x30
0x0D	Configuration 1 String (3)	Address location for iConfiguration string (divided by 2). This index must be set to 0x00 if the string is not present.	0x37
0x0E	Interface 1 String (4)	Address location for iInterface string (divided by 2). This index must be set to 0x00 if the string is not present.	0x3F
0x0F	Serial Number String (5)	Address location for iSerialNumber string (divided by 2). This index must be set to 0x00 if the string is not present. The *USB Mass Storage Class Bulk Only Transport Specification* requires a unique serial number.	0x4C
0x10	Configuration 2 String (7)	Address location for second iConfiguration 2 string (divided by 2). This index must be set to 0x00 if the string is not present.	0x61
0x11	Interface 2 String (8)	Address location for second iInterface 2 string (divided by 2).	0x6B
0x12	bLength	Length of device descriptor in bytes.	0x12
0x13	bDescriptor Type	Descriptor type.	0x01
0x14	bcdUSB (LSB)	USB Specification release number in BCD.	0x10
0x15	bcdUSB (MSB)		0x01
0x16	bDeviceClass	Device class.	0x00
0x17	bDeviceSubClass	Device subclass.	0x00
0x18	bDeviceProtocol	Device protocol.	0x00
0x19	bMaxPacketSize	Maximum USB packet size supported	0x40
0x1A	idVendor (LSB)	Vendor ID.	0xAB
0x1B	idVendor (MSB)		0x05
0x1C	idProduct (LSB)	Product ID.	0x30
0x1D	idProduct (MSB)		0x00
0x1E	bcdDevice (LSB)	Device release number in BCD.	0x00
0x1F	bcdDevice (MSB)		0x01
0x20	iManufacturer	Index to manufacturer string.	0x01
0x21	iProduct	Index to product string.	0x02
0x22	iSerialNumber	Index to serial number string.	0x05
0x23	bNumConfigurations	Number of configurations supported.	0x01
0x24	bConfiguration Value 2	The value to use as an argument to Set Configuration to select the configuration.	0x00
0x25	iConfiguration 2	Index to second configuration string.	0x00
0x26	bMaxPower 2	Maximum power consumption for the second configuration.	0x00
0x27	iInterface 2	Index to interface string associated with the second configuration descriptor.	0x00
0x28	bLength	Length of configuration descriptor in bytes.	0x09
0x29	bDescriptorType	Descriptor type.	0x02
0x2A	bTotalLength (LSB)	Number of bytes returned in this configuration. This includes the configuration descriptor plus all the interface and endpoint descriptors.	0x27
0x2B	bTotalLength (MSB)		0x00
0x2C	bNumInterfaces	Number of interfaces supported.	0x01
0x2D	bConfiguration Value 1	The value to use as an argument to Set Configuration to select the configuration.	0x01
0x2E	iConfiguration 1	Index to first configuration string.	0x00
0x2F	bmAttributes	Device attributes for this configuration.	0xE0
0x30	bMaxPower 1	Maximum power consumption for the second configuration.	0x31
0x31	bLength	Length of interface descriptor in bytes.	0x09
0x32	bDescriptorType	Descriptor type.	0x04
0x33	bInterfaceNumber	Interface number.	0x00
0x34	bAlternateSettings	Alternate settings	0x00
0x35	bNumEndpoints	Number of endpoints	0x03
0x36	bInterfaceClass	Interface class.	0x08

Address	Field Name	Description	Example SROM / FBh Data
0x37	bInterfaceSubClass	Interface subclass.	0x06
0x38	bInterfaceProtocol	Interface protocol.	0x50
0x39	iInterface 1	Index to interface string associated with the first configuration descriptor.	0x00
0x3A	bLength	Length of this descriptor in bytes.	0x07
0x3B	bDescriptorType	Endpoint descriptor type.	0x05
0x3C	bEndpointAddress	This is an Out endpoint, endpoint number 1.	0x01
0x3D	bmAttributes	This is a bulk endpoint.	0x02
0x3E	wMaxPacketSize (LSB)	Max data transfer size.	0x40
0x3F	wMaxPacketSize (MSB)		0x00
0x40	bInterval	Does not apply to bulk endpoints.	0x00
0x41	bLength	Length of this descriptor in bytes.	0x07
0x42	bDescriptorType	Endpoint descriptor type.	0x05
0x43	bEndpointAddress	This is an In endpoint, endpoint number 2.	0x82
0x44	bmAttributes	This is a bulk endpoint.	0x02
0x45	wMaxPacketSize (LSB)	Max data transfer size.	0x40
0x46	wMaxPacketSize (MSB)		0x00
0x47	bInterval	Does not apply to bulk endpoints.	0x00
0x48	bLength	Length of this descriptor in bytes.	0x07
0x49	bDescriptorType	Endpoint descriptor type.	0x05
0x4A	bEndpointAddress	This is an Interrupt endpoint, endpoint number 3.	0x83
0x4B	bmAttributes	This is an interrupt endpoint.	0x03
0x4C	wMaxPacketSize (LSB)	Max data transfer size.	0x02
0x4D	wMaxPacketSize (MSB)		0x00
0x4E	bInterval	This is the polling interval.	0x02
0x4F	Not used	Not used since starting address must be even.	0x00
0x50	bLength	LANGID string descriptor length in bytes.	0x04
0x51	bDescriptorType	Descriptor type.	0x03
0x52	LANGID (LSB)	Language supported.	0x09
0x53	LANGID (MSB)		0x04
0x54	bLength	Manufacturer string descriptor length in bytes (restricted to 63 bytes or less).	0x0C
0x55	bDescriptorType	Descriptor type.	0x03
0x56	bString	“M”	0x4D
0x57	bString	“NUL”	0x00
0x58	bString	“f”	0x66
0x59	bString	“NUL”	0x00
0x5A	bString	“g”	0x67
0x5B	bString	“NUL”	0x00
0x5C	bString	“ ”	0x20
0x5D	bString	“NUL”	0x00
0x5E	bString	“X”	0x58
0x5F	bString	“NUL”	0x00
0x60	bLength	Product string descriptor length in bytes (restricted to 63 bytes or less).	0x0E
0x61	bDescriptorType	Descriptor type	0x03
0x62	bString	“P”	0x50
0x63	bString	“NUL”	0x00
0x64	bString	“r”	0x72
0x65	bString	“NUL”	0x00

Address	Field Name	Description	Example SROM / FBh Data
0x66	bString	“o”	0x6F
0x67	bString	“NUL”	0x00
0x68	bString	“d”	0x64
0x69	bString	“NUL”	0x00
0x6A	bString	“ ”	0x20
0x6B	bString	“NUL”	0x00
0x6C	bString	“Y”	0x59
0x6D	bString	“NUL”	0x00
0x6E	bLength	Configuration 1 string descriptor length in bytes (restricted to 63 bytes or less).	0x0E
0x6F	bDescriptorType	Descriptor type	0x03
0x70	bString	“D”	0x44
0x71	bString	“NUL”	0x00
0x72	bString	“e”	0x65
0x73	bString	“NUL”	0x00
0x74	bString	“f”	0x66
0x75	bString	“NUL”	0x00
0x76	bString	“a”	0x61
0x77	bString	“NUL”	0x00
0x78	bString	“u”	0x75
0x79	bString	“NUL”	0x00
0x7A	bString	“l”	0x6C
0x7B	bString	“NUL”	0x00
0x7C	bString	“t”	0x74
0x7D	bString	“NUL”	0x00
0x7E	bLength	Interface 1 (Configuration 1) string descriptor length in bytes (restricted to 63 bytes or less).	0x1A
0x7F	bDescriptorType	Descriptor type	0x03
0x80	bString	“M”	0x4D
0x81	bString	“NUL”	0x00
0x82	bString	“a”	0x61
0x83	bString	“NUL”	0x00
0x84	bString	“s”	0x73
0x85	bString	“NUL”	0x00
0x86	bString	“s”	0x73
0x87	bString	“NUL”	0x00
0x88	bString	“ ”	0x20
0x89	bString	“NUL”	0x00
0x8A	bString	“S”	0x53
0x8B	bString	“NUL”	0x00
0x8C	bString	“t”	0x74
0x8D	bString	“NUL”	0x00
0x8E	bString	“o”	0x6F
0x8F	bString	“NUL”	0x00
0x90	bString	“r”	0x72
0x91	bString	“NUL”	0x00
0x92	bString	“a”	0x61
0x93	bString	“NUL”	0x00

Address	Field Name	Description	Example SROM / FBh Data
0x94	bString	“g”	0x67
0x95	bString	“NUL”	0x00
0x96	bString	“e”	0x65
0x97	bString	“NUL”	0x00
0x98	bLength	Serial Number string descriptor length in bytes (restricted to 63 bytes or less).	0x2A
0x99	bDescriptorType	Descriptor type	0x03
0x9A	bString	“0”	0x30
0x9B	bString	“NUL”	0x00
0x9C	bString	“1”	0x31
0x9D	bString	“NUL”	0x00
0x9E	bString	“2”	0x32
0x9F	bString	“NUL”	0x00
0xA0	bString	“3”	0x33
0xA1	bString	“NUL”	0x00
0xA2	bString	“4”	0x34
0xA3	bString	“NUL”	0x00
0xA4	bString	“5”	0x35
0xA5	bString	“NUL”	0x00
0xA6	bString	“6”	0x36
0xA7	bString	“NUL”	0x00
0xA8	bString	“7”	0x37
0xA9	bString	“NUL”	0x00
0xAA	bString	“8”	0x38
0xAB	bString	“NUL”	0x00
0xAC	bString	“9”	0x39
0xAD	bString	“NUL”	0x00
0xAE	bString	“0”	0x30
0xAF	bString	“NUL”	0x00
0xB0	bString	“1”	0x31
0xB1	bString	“NUL”	0x00
0xB2	bString	“2”	0x32
0xB3	bString	“NUL”	0x00
0xB4	bString	“3”	0x33
0xB5	bString	“NUL”	0x00
0xB6	bString	“4”	0x34
0xB7	bString	“NUL”	0x00
0xB8	bString	“5”	0x35
0xB9	bString	“NUL”	0x00
0xBA	bString	“6”	0x36
0xBB	bString	“NUL”	0x00
0xBC	bString	“7”	0x37
0xBD	bString	“NUL”	0x00
0xBE	bString	“8”	0x38
0xBF	bString	“NUL”	0x00
0xC0	bString	“9”	0x39
0xC1	bString	“NUL”	0x00

Address	Field Name	Description	Example SROM / FBh Data
0xC2	bLength	Configuration 2 string descriptor length in bytes (restricted to 63 bytes or less).	0x0E
0xC3	bDescriptorType	Descriptor type	0x03
0xC4	bString	“L”	0x4C
0xC5	bString	“NUL”	0x00
0xC6	bString	“o”	0x6F
0xC7	bString	“NUL”	0x00
0xC8	bString	“w”	0x77
0xC9	bString	“NUL”	0x00
0xCA	bString	“ ”	0x20
0xCB	bString	“NUL”	0x00
0xCC	bString	“P”	0x50
0xCD	bString	“NUL”	0x00
0xCE	bString	“o”	0x6F
0xCF	bString	“NUL”	0x00
0xD0	bString	“w”	0x77
0xD1	bString	“NUL”	0x00
0xD2	bString	“e”	0x65
0xD3	bString	“NUL”	0x00
0xD4	bString	“r”	0x72
0xD5	bString	“NUL”	0x00
0xD6	bLength	Interface 2 (Configuration 2) string descriptor length in bytes (restricted to 63 bytes or less).	0x0E
0xD7	bDescriptorType	Descriptor type	0x03
0xD8	bString	“B”	0x42
0xD9	bString	“NUL”	0x00
0xDA	bString	“u”	0x75
0xDB	bString	“NUL”	0x00
0xDC	bString	“s”	0x73
0xDD	bString	“NUL”	0x00
0xDE	bString	“ ”	0x20
0xDF	bString	“NUL”	0x00
0xE0	bString	“P”	0x50
0xE1	bString	“NUL”	0x00
0xE2	bString	“o”	0x6F
0xE3	bString	“NUL”	0x00
0xE4	bString	“w”	0x77
0xE5	bString	“NUL”	0x00
0xE6	bString	“e”	0x65
0xE7	bString	“NUL”	0x00
0xE8	bString	“r”	0x72
0xE9	bString	“NUL”	0x00
0xEA	bString	“e”	0x65
0xEB	bString	“NUL”	0x00
0xEC	bString	“d”	0x64
0xED	bString	“NUL”	0x00

Address	Field Name	Description	Example SROM / FBh Data
0xEE – 0x1FF (SROM), 0x13F (FBh)	Not used	Available SROM space / unused FBh space	0xXX

Table 20 – Example Serial ROM / FBh Data