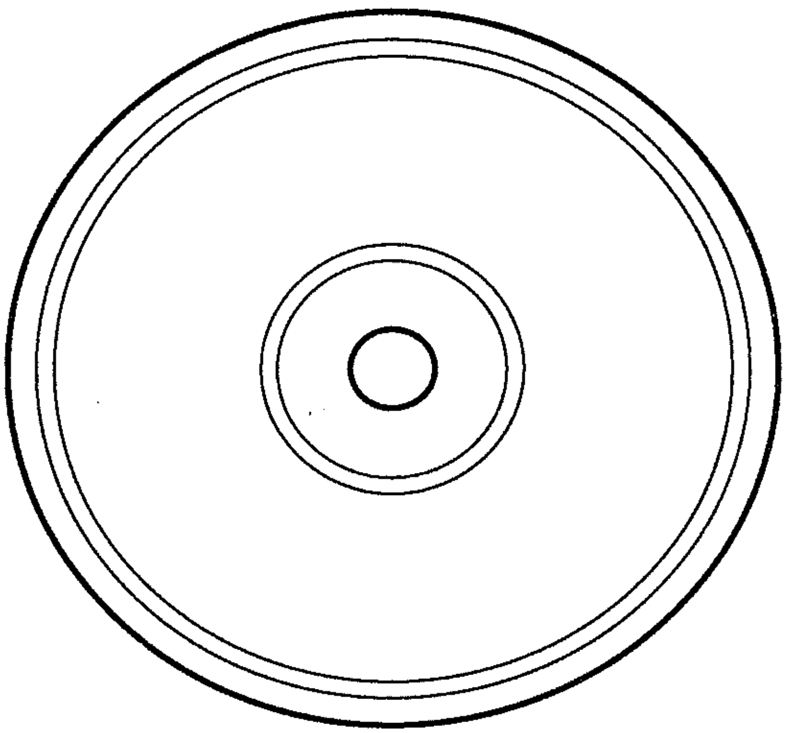
CONFIDENTIAL

SUPER NES CD-ROM SYSTEM

CD-ROM OVERVIEW



System Configuration

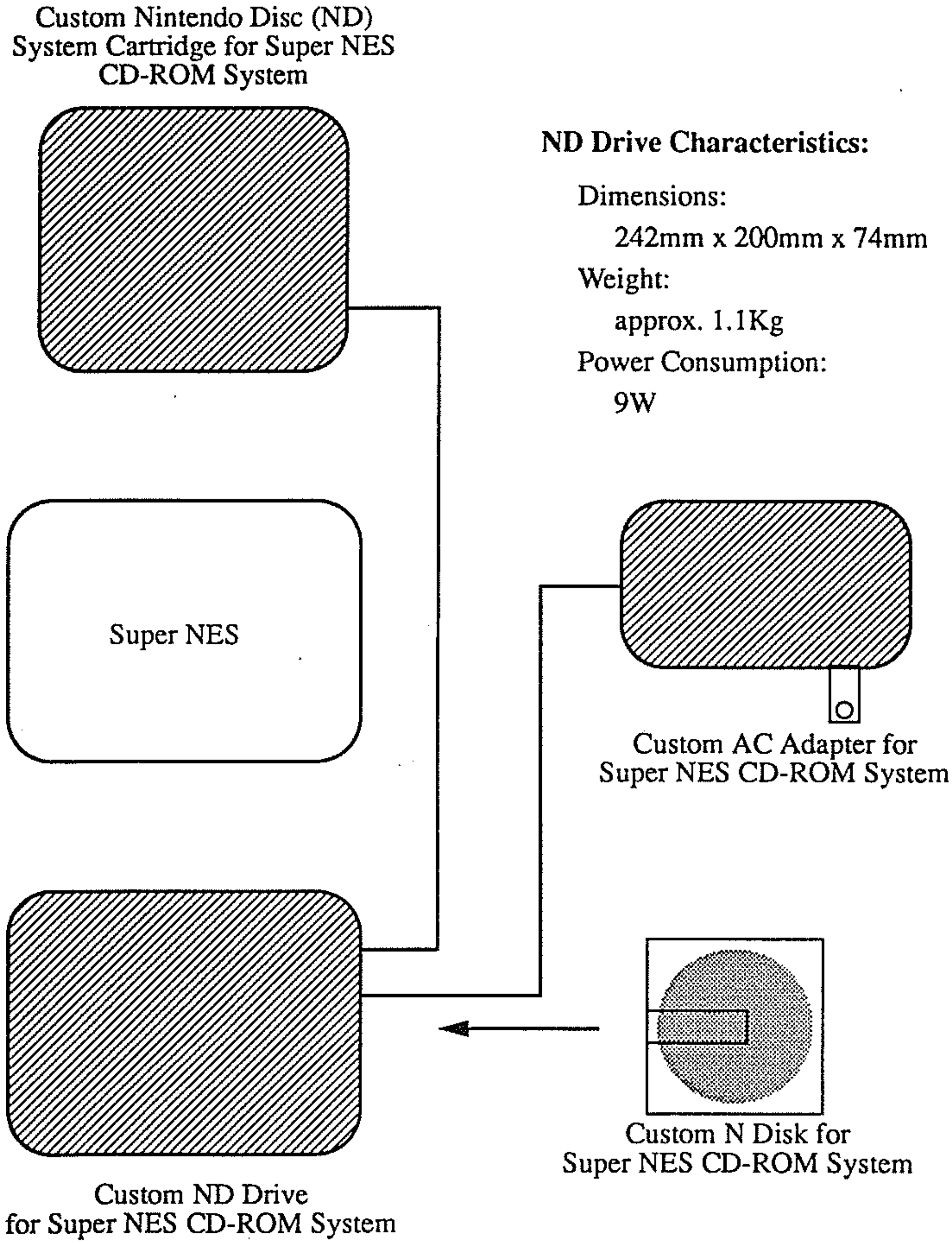
1.1 SUPER NES CD-ROM SYSTEM CONFIGURATION

ND Drive Characteristics:

Dimensions:
242mm x 200mm x 74mm

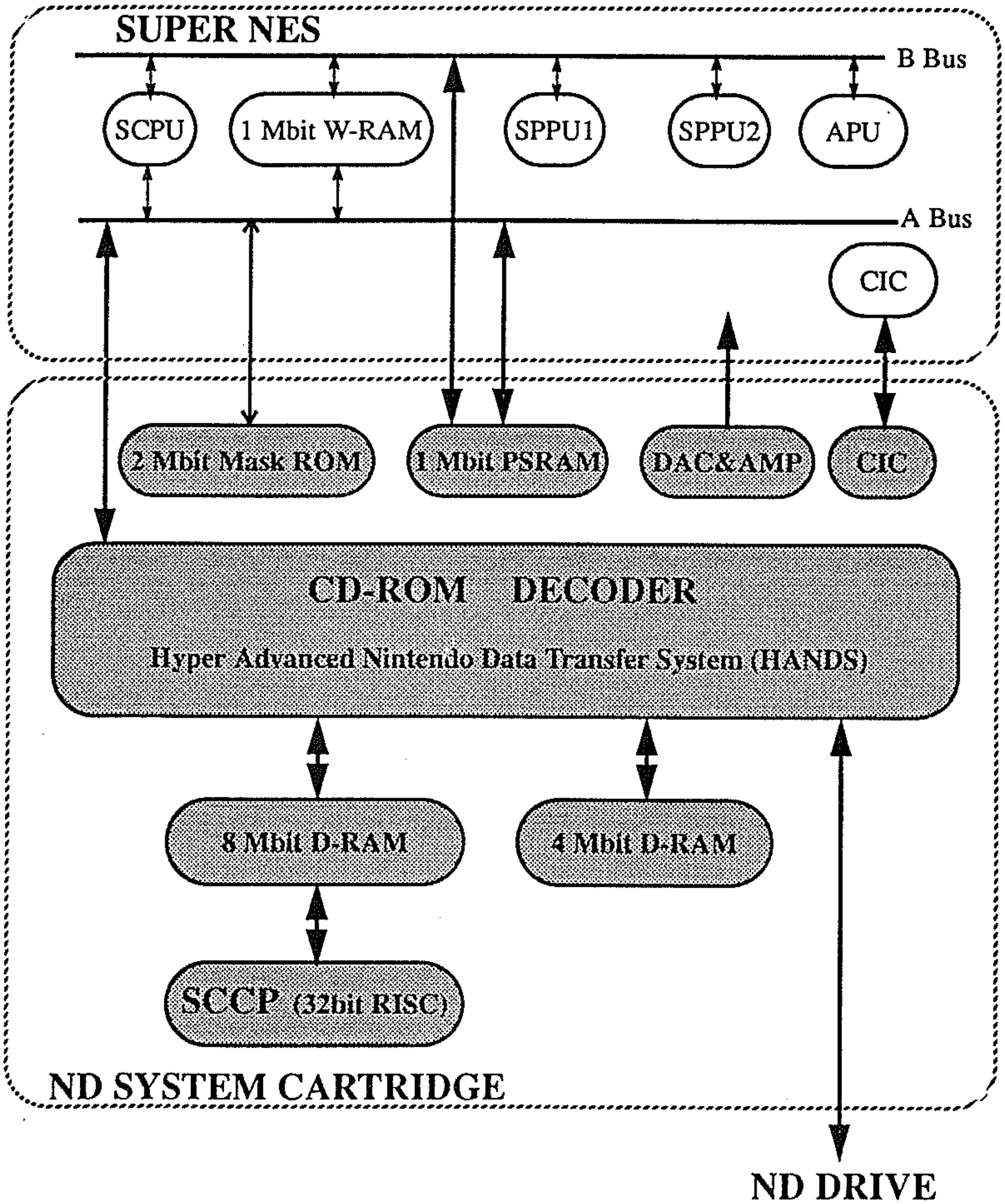
Weight:
approx. 1.1Kg

Power Consumption:
9W



▨ These devices will be sold as a CD-ROM Adapter kit.

2.1 SYSTEM BLOCK DIAGRAM (Software Perspective)



2.2 MEMORY CONFIGURATION

1. Main Memory

• D-RAM	8 Mbits
• PS-RAM	1 Mbit
• (Super NES Memory	1 Mbit)

2. Supplementary Memory

• D-RAM	4 Mbits

3. System ROM 2 Mbits

2.3 CO-PROCESSOR (SCCP)

The Super NES CD-ROM System Co-processor (SCCP) is a 32-bit high-speed RISC micro-processor, designed to drastically improve the Super NES CD-ROM System performance.

1. Core — 32-bit RISC CPU
2. Clock — 21.47727 MHz no-wait
3. Cache — Command cache (1 Kbyte)
4. Arithmetic Functions

   Powerful arithmetic functions including; floating-point arithmetic commands, addition, subtraction, multiplication, and division operations

5. Parallel Operation with the Super NES CPU
6. Program Development Environment

   Environment using a highly optimized C compiler

7. Examples of Application
   - Graphics image processing
   - Image data compression and expansion
   - Hit detection processing
   - Logic program

2.4 CD-ROM DECODER (HANDS)

The Hyper Advanced Nintendo Data Transfer System (HANDS) is a custom chip with functions which allow efficient utilization of the Super NES CD-ROM System's large memory capacity and XA-compatible CD-ROM decoding functions.

1. Core 65C02

2. Clock 4.29545 MHz

3. DMA Support Functions

   The following two functions support Super NES CPU DMA.

   a. HANDS allows the data on the main memory (8-Mbit DRAM) to be read without stopping the SCCP.

   b. When the SCCP DMA-transfers graphics data contained within memory to the VRAM, the graphics data is automatically converted to character data format for the Super NES.

4. ADPCM Decoder

   HANDS features a sound map function, which allows output of ADPCM audio data stored in memory using any timing scheme.

   HANDS includes two units of ADPCM decoders (one unit = two channels). This makes it possible to output the sound map ADPCM audio while playing the CD ADPCM audio.

Drive and Media Specifications

3.1 DRIVE SPECIFICATIONS

1. Data Output Speed	(Normal)	approx. 150 Kbytes/sec.
	(Double)	approx. 300 Kbytes/sec.
2. Standard Access Time (1/3 stroke)		0.7 sec.
3. Maximum Access Time (full stroke)		1.4 sec.
4. Read Error Rate		10^{-12} or less

3.2 MEDIA SPECIFICATIONS

1. CD-ROM XA Format
2. Maximum Data Capacity while Formatting approx. 540 Mbytes
3. Custom Case

The disk is sealed in a rugged cartridge, designed to reduce the chance of inadvertent damage through consumer handling.

The case contains a software security device which makes illegal duplication of the contained software difficult.

The custom case also contains a back-up RAM for game data storage. A capacity of 256-Kbit is currently planned for this RAM.

4.1 FUTURE DEVELOPMENT TOOLS

1. CD-ROM Drive Emulator (CD-BOX)

   The CD-BOX is a CD-ROM drive emulator developed for the Super NES CD-ROM System. The CD-ROM drive emulation function, essential in the development of CD-ROM software, allows debugging. Images and sound can be output with the same characteristics they will have when output from the CD.

   The CD-BOX contains a CD-image builder, PCM-sound-read, and ADPCM-conversion tools.

   The CD-BOX supports CD-WO write and master tape creation functions for production of the CD using the emulator, after debugging is completed.

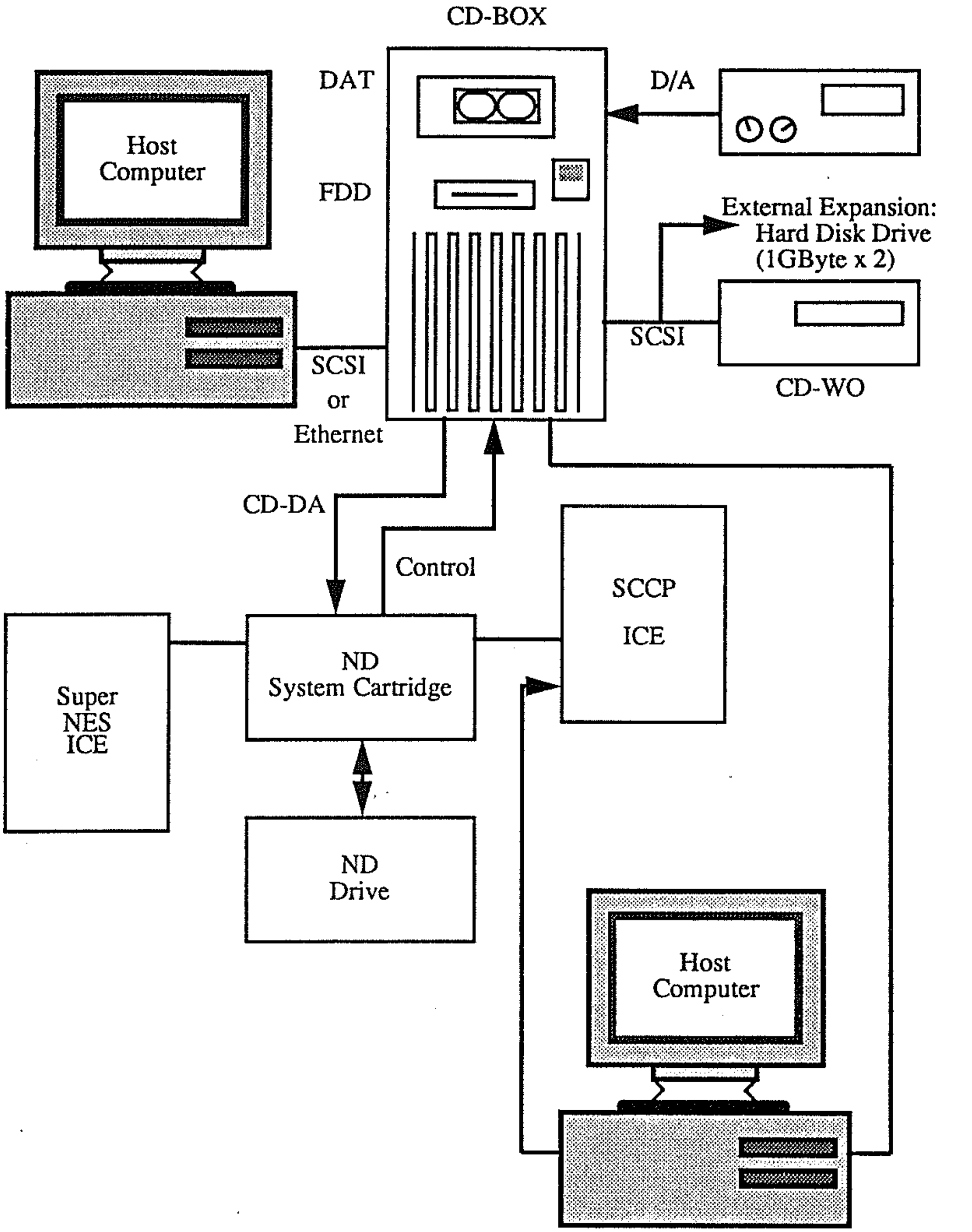
2. SCCP Program Development Environment

   A C-compiler package (C compiler, assembler, linker, and archiver) and custom debuggers (ICE, and source debugger) are used.

3. Real-Time File Editor

   Graphic interface software is used for command output and file transfer to the CD-BOX. This software provides real-time file editing functions.

4.2 HARDWARE CONFIGURATION



Super NES CD-ROM System Overview

5.1 MEMORY CONFIGURATION

1. Main Memory
   - D-RAM — 8 Mbits
   - PS-RAM — 1 Mbit

   NOTE: The Super NES also provides 1 Mbit of usable memory to CD-ROM.

2. Supplementary Memory
   - D-RAM — 4 Mbits

3. System ROM — 2 Mbits

5.2 SCCP

1. Core	32-bit RISC CPU
2. Clock	21.47727 MHz no-wait
3. Cache	Command cache (1 Kbyte)

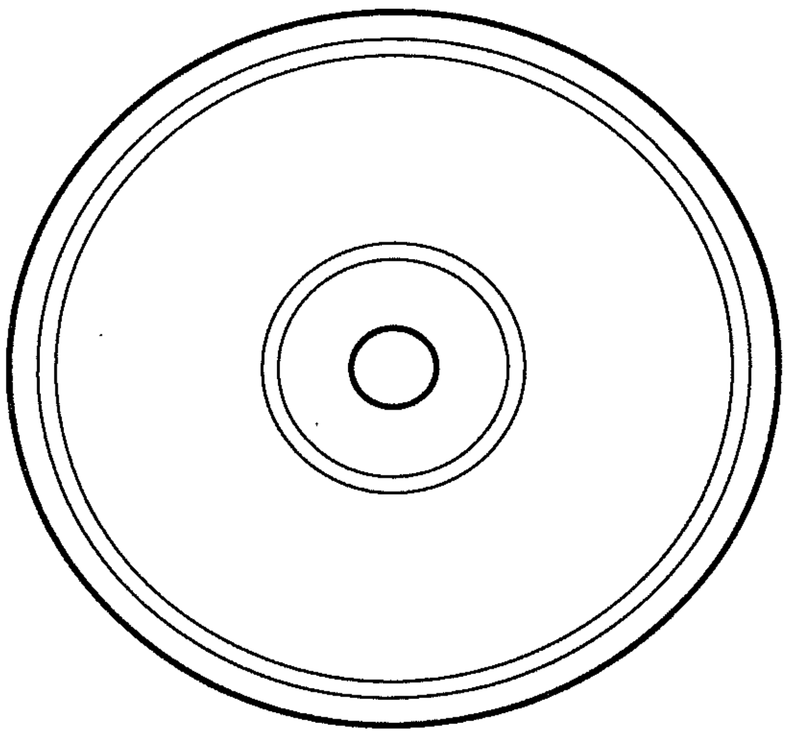
5.3 DRIVE SPECIFICATIONS

1. Data Output Speed	(Normal)	approx. 150 Kbytes/sec.
	(Double)	approx. 300 Kbytes/sec.
2. Standard Access Time (1/3 stroke)		0.7 sec.
3. Maximum Access Time (full stroke)		1.4 sec.
4. Read Error Rate		10^{-12} or less

CONFIDENTIAL

SUPER NES CD-ROM SYSTEM

CD-ROM TUTORIAL



Nintendo Technical Conference of February 1, 1993

CD-ROM Basics

This chapter provides a basic introduction to CD-ROM for software developers.

1.1 WHAT IS CD-ROM?

1.1.1 CD-ROM

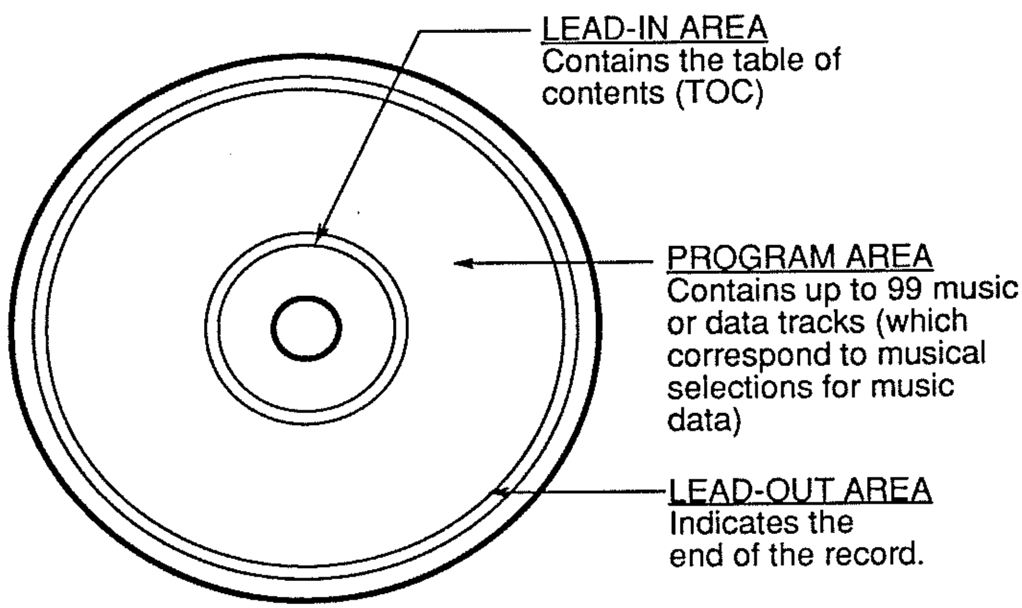
CD-ROM is a read-only compact disc (CD) encoded with computer data. CDs were initially developed for audio applications. CD audio discs and players can be found in many households today. How they work, however, is often not understood.

1.1.2 COMPACT DISCS

CDs measure 12 cm in diameter. Data are represented as microscopic pits on a single path which spirals outward from the inner radius of the disc. A laser beam is used to detect these pits. The optical interference which results when light shines on the pits gives the disc a rainbow-colored, shimmering appearance.

1.1.3 DISC STRUCTURE

The disc is divided into three areas, as shown below.



1.2 MAJOR FEATURES OF CD-ROM

1.2.1 LARGE STORAGE CAPACITY

Good software design must anticipate the ability of CDs to hold large volumes of data. The average CD-ROM holds about 540 Mbytes of data. Up to 650 Mbytes can be recorded within CD-ROM standards.

The 540 Mbytes is 500 times greater than 8 Mbits of semi-conductor ROM.

1.2.2 DATA ACCESS TIME

Most people assume that CD games are slow. Access time is relatively fast when compared to other large capacity memory devices, but is very slow compared to semiconductor ROM.

CDs utilize a rotation technique called constant linear velocity (CLV) to maximize the recording density on a disc. Under CLV, the inner radius of the disc spins more rapidly than the outer radius. When moving between sectors on the inner and outer radius of the disk, the head must move a long distance and the motor speed changes. This increases the seek time. Adjacent positions on the disc, however, can be rapidly accessed by slightly shifting the direction of the laser beam.

A CD's slow access time may also be attributed to rotational delay. The disc must rotate from the time the read head settles into position until the disc rotates to the desired sector.

Effective CD-ROM programs must take these limitations into account. Slow CD games cannot be solely attributed to the medium. Half of the blame belongs to poor programming techniques.

But, relax! The Nintendo CD-ROM accesses and reads data while the program is being executed. This helps conceal the drawbacks of CD technology.

1.2.3 DATA TRANSFER RATE

The slow seek speed of CDs is only one reason CD games may be slow. Unlike semiconductor ROM, CD-ROM data must first be read into RAM in order to be used. The CD-ROM data transfer rate is 150 Kbytes/second, under the Mode 2 Form 1 recording format, and 170.2 Kbytes/second, under the Mode 2 Form 2 format.

This data transfer rate must be considered when designing animation software, as screen and sound data must be read in real-time.

1.3 ERROR CORRECTION

1.3.1 CD-ROM ERROR CORRECTION

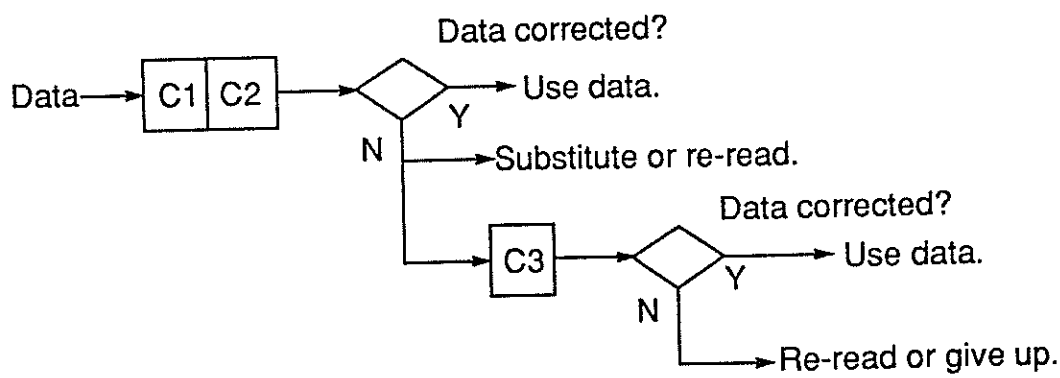
Due to their high data density, CDs are highly susceptible to errors from damage incurred during the disc mastering and production processes and user handling. CDs use an error correction code called the Cross Interleave Reed-Solomon Code (CIRC). CIRC employs two independent Reed-Solomon codes, which are referred to as C1 and C2. The error correction rate under C1 and C2 is not perfect, however, and only a certain range of errors can be corrected. When non-correctable errors occur, either another technique must be employed to compensate for the data or the data cannot be used.

For CD audio, other data can be interpolated for erroneous data without loss of audio fidelity. This is not permissible for computer data or text, therefore, CD-ROM uses an additional level of error correction. The third level is called C3. When error correction using C1 and C2 is not successful, C3 is utilized. Together, the three layers of error correction achieve a very low rate of non-correctable data.

1.3.2 WHEN ERRORS OCCUR

Hardware automatically implements C1 and C2 when data is read from the disc. Most errors are corrected by these codes. When C1 and C2 are unsuccessful, the program is informed of the position of the data error, and the program either identifies a substitute for the data or re-reads the data (when redundant data is appended to the original).

The final level of error correction, C3, is a software routine implemented for program and other data which must meet stringent data integrity standards. The C3 correction routine is called by software, when required.



1.4 SUBCODES

1.4.1 SUBCODES

Subcodes are information recorded intermittently among the actual data on a CD. Subcodes serve two purposes:

1. To search for a song or start playing from a certain point.
2. To allow information other than sound data to be recorded, such as text or graphics which will be displayed on the screen.

The two subcode channels P and Q are used for the first of these purposes. The six subcode channels R, S, T, U, V, and W are used for the second purpose. The standards for these channels were set with CD audio in mind, therefore, data cannot be freely recorded and used as desired.

Channels R through W record screen data. The format for these channels is determined in the CD Graphics Standards (CD-G). The most successful use of these channels is for Japanese *"karaoke"*, which displays the words to a song beneath a still picture.

The P channel is a music track separator flag. When this channel is set to 1, the drive pauses between songs.

The Q channel is used for finer control purposes and contains information concerning the absolute (real) time and elapsed time within a song track. The Q channel in the lead-in area is used for the table of contents (TOC).

1.4.2 DISC TABLE OF CONTENTS (TOC)

The TOC is recorded in the lead-in area on a CD and contains a list which identifies the track for each song. The TOC is essential for CD audio players and CD-DA standards require that a TOC always be recorded.

Data on CD-ROM, however, is accessed by files rather than by tracks, and programs do not generally need a TOC.

The CD Family

The CD was initially developed for audio, but has since been adapted for a variety of applications. The type of CD to be developed by Nintendo is called CD-ROM XA. This chapter reviews the history of the CD, as a basis for understanding CD-ROM XA.

2.1 FROM CD-DA TO CD-ROM

2.1.1 THE HISTORY OF THE CD

CD technology was originally developed for audio applications. The standard developed for audio discs is the "Compact Disc Digital Audio Specifications", or more commonly called the Red Book. The Red Book specifies the physical format of CDs, the width of the spiral grove on which audio data is recorded, and so on. The Red Book is the basis for all other CD specifications.

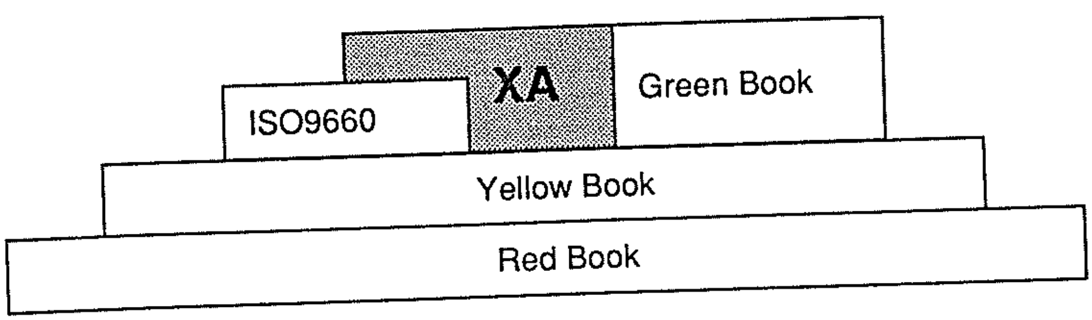
Next, the Yellow Book, or "CD-ROM Specifications", was released to specify the physical format for recording computer data on a CD. This standard specifies the physical arrangement of data, but does not define how data should be organized into files. It leaves the logical format of each CD-ROM up to the discretion of the developer.

The International Standard Organization's ISO 9660 was subsequently developed to specify the logical format of the file structure for CD-ROM. Most CD-ROM software on the market today conform to this standard.

In addition, standards for mass-market interactive CD media were defined in the "Compact Disc Interactive (CD-I) Specifications" - the Green Book. CD-I incorporates newer technologies, such as interleaving sound and graphics.

2.1.2 THE RELATIONSHIP BETWEEN THE VARIOUS CD STANDARDS

The relationships between these standards are indicated in the figure below.



2.2 CD-DA FOR MUSIC

2.2.1 PHYSICAL FORMAT OF THE DISC

The Red Book defines standards for the mechanical and optical characteristics of the CD-DA disc. It specifies pulse code modulation (PCM) as the modulation technique used to digitalize audio signals and the Cross Interleave Reed-Solomon Code (CIRC) as the error correction code.

2.2.2 AUDIO REPLAY

The stereo PCM sound on a CD-DA disc uses a sampling frequency of 44.1 kHz and 16 quantization bits. Up to 99 tracks can be recorded in the program area. Each track holds one musical selection. The table of contents in the lead-in area lists the minute and second at which each song track begins.

The timing information contained in the subcode Q channel is needed to replay CD-DA sound. The Q channel information, however, is only accurate within $\pm$ 1 second, due to errors which may occur during the mastering process. This accuracy rate is inadequate for replaying certain sounds within a track, such as sound effects in a game. A game user will readily notice a sound effect which is not replayed until one second after the corresponding visual event has occurred.

CD-DA sound is used for little more than background music in CD games for the above-mentioned reasons and because the transfer rate does not allow non-audio data to be read during audio replay.

2.3 CD-ROM FOR COMPUTER DATA

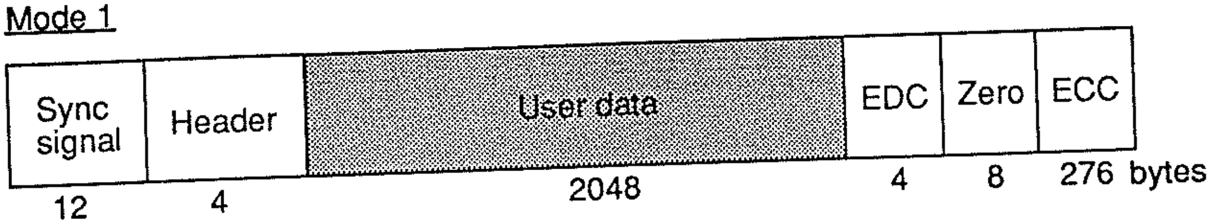
2.3.1 CD-ROM DISCS

A CD-ROM disc includes a data track on which computer data is recorded and an audio track on which CD-DA sound is recorded. A combined total of up to 99 tracks can be recorded on the disc.

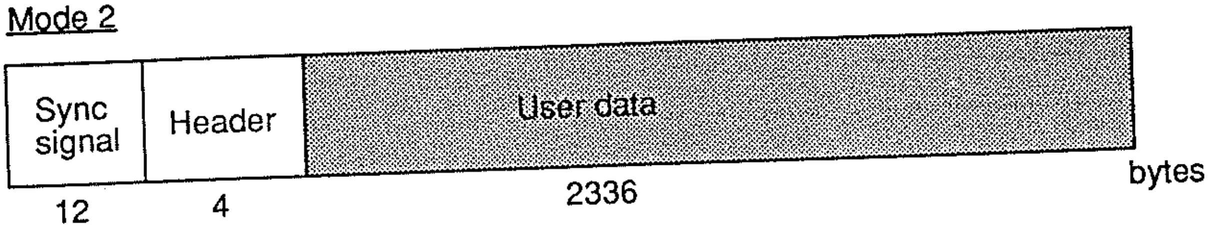
2.3.2 DISC FORMAT

The Yellow Book specifies the physical format of data recorded on the disc. The smallest units which can be physically accessed on the disc are called sectors. One sector is 2352 bytes. As indicated in the figure below, a CD-ROM sector can be recorded in either Mode 1 or Mode 2.

Mode 1



Mode 2



A fixed pattern is recorded in the first 12 bytes of every sector for synchronization. The next 4 bytes are the header field. The header contains the absolute time information for the sector (i.e., the sector address) and one byte which indicates if the sector is recorded in Mode 1 or Mode 2. The absolute time information is used to access each sector independently.

Mode 1 is used to record data which require a very low data error rate. Mode 1 contains an error detection code (EDC) and error correction code (ECC) which may be applied within a sector in addition to the CIRC for the disc. ECC is referred to as C3.

Mode 2 allows 2336 bytes of user data to be recorded and has the same level of error correction as a CD-DA disc.

2.4 CD-I DISC FOR THE MASS MARKET

2.4.1 CD INTERACTIVE

The CD-I Standard was created for consumer and educational uses of CD-ROM technology. CD-I is more oriented toward multi-media applications than CD-ROM and allows the user to use audio and video interactively.

CD-ROM discs store video and audio data on separate tracks (called, respectively, data and audio tracks). To replay video and audio information simultaneously, the video data on the data track is read and stored in memory first. Then, the audio track is read and replayed at the same time the video data is displayed. To display the next screen, the data track is accessed again, but audio replay must be suspended during this period. This format is obviously inadequate for fully interactive, multi-media applications.

To alleviate this problem, CD-I allows audio data to be compressed and encoded, using Adaptive Delta Pulse Code Modulation (ADPCM), and interleaves audio and video data on one track. This new technology, introduced in CD-I, is also incorporated into CD-ROM XA.

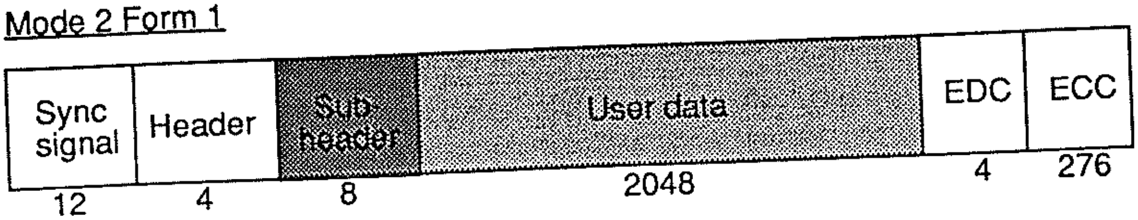
2.5 CD-ROM XA

2.5.1 CD-ROM XA DISC FORMAT

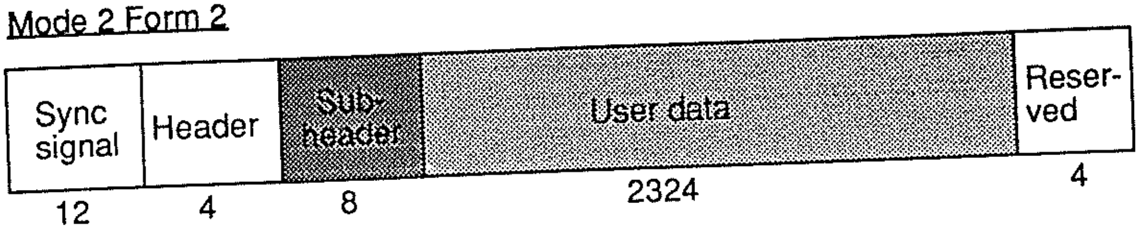
CD-ROM XA improves CD-ROM's multimedia potential by incorporating CD-I specifications. With the exception of the file structure, CD-ROM XA is nearly identical to CD-I.

The CD-ROM XA disc format includes a subheader not found in CD-ROM and divides CD-ROM Mode 2 into two forms; Form 1 and Form 2.

Mode 2 Form 1



Mode 2 Form 2



The subheader allows CD-ROM XA to read interleaved audio and video files in real-time.

2.5.2 FILE STRUCTURE

The data on a CD-ROM XA disc is divided into, and accessed as, files. This file structure is based on ISO 9660. Files are managed using a hierarchical directory structure similar to that found on MS-DOS and UNIX-based systems. CD-ROM XA's ability to process data in file units improves the ease of development.

2.5.3 ADAPTIVE DELTA PULSE CODE MODULATION (ADPCM)

An additional feature of CD-ROM XA is its use of ADPCM audio. ADPCM enables four to sixteen times more compressed audio data to be stored on a CD-ROM XA disc than can be stored using PCM on a CD-DA disc. ADPCM allows video data to be read and audio data replayed simultaneously.

CD-ROM XA

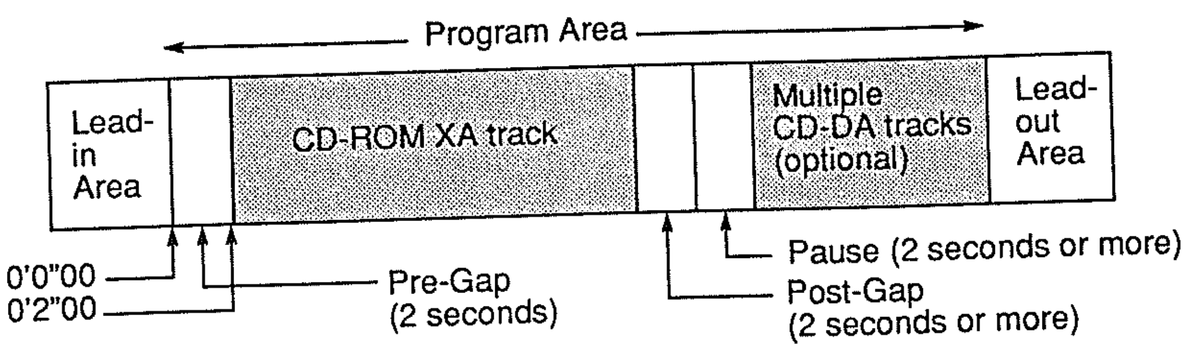
This chapter describes the disc structure, format, features, and file structure of CD-ROM XA. CD-ROM XA's disc structure must be understood before the Super NES CD-ROM system's call function can be fully utilized.

3.1 CD-ROM XA DISC STRUCTURE

3.1.1 THE CD-ROM XA DISC

A CD-ROM XA disc can hold up to 99 tracks, including data tracks (XA tracks) containing computer data and audio tracks containing CD-DA sound. CD-DA audio tracks are optional. However, the track order is strictly specified. XA tracks must always be placed before CD-DA tracks. As recommended in the CD-ROM XA Standards, the Nintendo disc uses only one XA track. Data is accessed by files, not by tracks as in CD-DA format.

As shown in the following figure, XA tracks are proceeded by a gap of two or more seconds, while XA and CD-DA tracks are separated by a gap and a pause of two or more seconds each.



3.1.2 THE XA TRACK

The XA track is divided into 2352-byte Mode 2 Form 1 and Mode 2 Form 2 sectors. Sectors are the smallest units on the disc which can be physically accessed. Sectors are identified by an address in the header field which includes bytes for minute, second, and sector. One second is comprised of 75 sectors and one minute equals 60 seconds. The header field identifies the minute, second, and sector of a desired sector on the disc.

This system is ideal for CD audio, but not for computer data. Therefore, under CD-ROM XA, logical sector numbers are assigned sequentially to sectors beginning with 0 at minute 0, second 2, and sector 00. For example, logical sector number 1000 is used to identify a desired sector. Data is processed as files in a normal program. One need not be aware of the absolute sector number for the data desired.

3.2 XA FORMAT

3.2.1 XA SECTOR FORMAT

Mode 2 Form 1

Sync signal	Header	Sub-header	User data	EDC	ECC
12	4	8	2048	4	276

Mode 2 Form 2

Sync signal	Header	Sub-header	User data	Reser-ved
12	4	8	2324	4

The XA track consists of 2352-byte Mode 2 Form 1 and Mode 2 Form 2 sectors.

Form 1 sectors include error correction codes (ECC) and are appropriate for data with high data-integrity requirements, such as computer data and screen data not correctable by interpolation. Software error correction, C3, is used to correct errors which C1 and C2 failed to correct in the hardware.

Form 2 sectors hold 2324 bytes of user data, and have the same level of error correction as CD audio. Form 2 sectors are appropriate for ADPCM audio data and screen data correctable by interpolation.

The fields in each mode are explained later.

3.2.2 SYNC AND HEADER FIELDS

The first 12 bytes of all sectors are recorded with a pre-set pattern for synchronization. This is the sync field. The next four bytes constitute the header field. The first three bytes of the header field hold the sector address, with one byte each for minute, second, and sector. The fourth byte indicates the mode used to record the sector.

3.2.3 SUBHEADER

This field is one of the distinguishing features of the XA standards. The submode field is eight bytes and consists of the following four one-byte items, which are double-written for data integrity.

3.2.3.1 File Number

Data is accessed in logical groups called "files" on an XA disc. Files consist of a multiple number of sectors.

An identical file number is assigned to all the sectors which belong to one file. When a file is read, only the sectors which contain that file number are read. The use of file numbers allows sectors from different files to be interleaved, i.e., physically alternated on the disc, without impairing the ability to read only the sectors in a particular file.

File numbers range from 0~255. File number zero is only assigned to files composed of sectors which must be read consecutively and not interleaved. File numbers 1~255 are used for files with either consecutive or interleaved sectors. A file number does not act as an absolute identifier for a file, as this would limit the number of files to 256, but as a means for discriminating between adjacent sectors.

Note that "interleaved files" are not the same thing as, what is generally referred to as, "interleaved audio and video". File interleaving uses a file number to enable the interleaving of two or more files. Interleaved audio and video refers to the alternation of audio and video sectors in one file. The distinction is evasive, but important. In general, "interleaving" refers to the interleaving of sectors in one file (sector interleaving).

Sector interleaving is sufficient for most requirements and file interleaving is relatively uncommon.

3.2.3.2 CHANNEL NUMBER

A channel number is used to identify several different pieces of information within a file that need to be chosen in combination. One example is a file containing narration in six languages. Channel numbers from 0~15 may be assigned to audio sectors (0~31 for video and data sectors). There are four types of sectors, which are explained in the next section.

3.2.3.3 SUBMODE

The submode defines the attributes of a sector. Attribute includes a sector's type. The four possible types of sectors are audio, video, data, and empty sector. An empty sector is a sector filled with zeros, used to fill gaps in a file. Hence, only three sectors are used for storing actual data: audio, video, and data sectors.

File number, submode, and channel number can be used to specify, for example, "play a certain channel of the ADPCM audio data in a file and store video and program data in certain memory locations", etc.

3.2.3.4 CODING INFORMATION

The coding information field defines the details about the type of data for each sector type. For example, the field could be used to define the ADPCM level for audio sectors.

3.3 XA FEATURES

3.3.1 THREE SECTOR TYPES + CD-DA

Sectors are one of three types, depending on the type of data recorded. These types are audio sectors containing ADPCM audio data, video sector containing video data, and data sectors containing program data. The distinction between video data and program data is ambiguous and the same sector may be designated as either a video or data sector depending on how the data in that sector is to be used.

In addition to these three types, CD-DA audio may also be recorded. These four types of data can be used creatively to develop interesting software.

3.3.2 ADPCM AUDIO

The XA format allows for improved audio. ADPCM audio will be used in addition to the current APU used by the Super NES. Of course, normal CD audio (CD-DA) can also be used. Specifications for the ADPCM defined in the XA standards are given in the following table.

	Sampling Frequency	Quantization Bit Number	Stereo/ Mono	Compression Rate
Level B	37.8 kHz	4	Stereo	4
			Mono	8
Level C	18.9 kHz	4	Stereo	8
			Mono	16

As indicated by the table, ADPCM audio is compressed 4~16 times more than PCM. This means that under Level B, for example, audio data which requires four sectors in CD-DA can be compressed into one sector, while the other sectors are used for video or text data. These sectors can then be incorporated into one file, so that the video data can be read at the same time the audio is replayed. This is an example of "interleaving".

The sound quality of Level B is equivalent to the quality of an FM broadcast and Level C is equivalent to an AM broadcast. Level B is appropriate for replaying orchestral music and Level C is adequate for narration and speech. Selecting the level which is adequate is important.

The sound quality is lower than on CD-DA. The advantages, however, are that interleaving is possible, the data recorded on XA sectors can be accessed without the ± 1 second timing error which occurs on CD-DA, and sound mapping is possible, i.e., the sound can be read into memory and played at a precise time.

3.3.3 REAL-TIME FILE READ

The XA subheader is used to simultaneously read a file in real-time, replay the ADPCM audio, and display the video. Unlike CD-ROM, CD-ROM XA does not require that video data be read and stored for subsequent replay with audio data. Rather, CD-ROM XA allows a long file to be read and used in a continuous sequence. This capability, referred to as a "stream", makes CD-ROM XA ideal for animation.

Data in the various sectors is routed into memory or an output device by specifying the subheader file number, channel number, and the sector type in the submode. For example, the programmer can specify, "play the second channel of ADPCM audio in a file, read the fifth channel of video memory into a certain memory location, read the program data into a certain memory location, and so on." Programs only need to display or use the data at the proper time.

Good program design requires working within the limitations of the data transfer rate. CDs transfer only 75 sectors per second. This equals 150 Kbytes in Mode 2 Form 1. All the necessary data must be read within this transfer rate.

3.4 XA FILE STRUCTURE

3.4.1 FILES

The average CD holds 540 Mbytes of data. Accessing the data by sequential sector numbers is inefficient. Also, if the order of data on the disc is modified, the program must also be changed. This is clearly unacceptable.

Hence, files are used to organize the sectors of data. Users of MS-DOS-based personal computers are already familiar with the concept of files. Files are logical collections of data, and data is accessed in file units. The program only needs to know which file and what portion of the file to read. It does not need to be aware of the sectors where the data is recorded on the disc. The tedious task of tracking down sectors is handled by the System Service Utility (SSU). The SSU supplied by Nintendo performs file management.

3.4.2 ACCESSING FILES

This section describes how the file manager determines where a desired file is recorded on the disc and retrieves its data. How the SSU accesses a particular file is determined by the CD-ROM XA standards. The CD-ROM XA file structure is based on the International Standard Organization's ISO 9660.

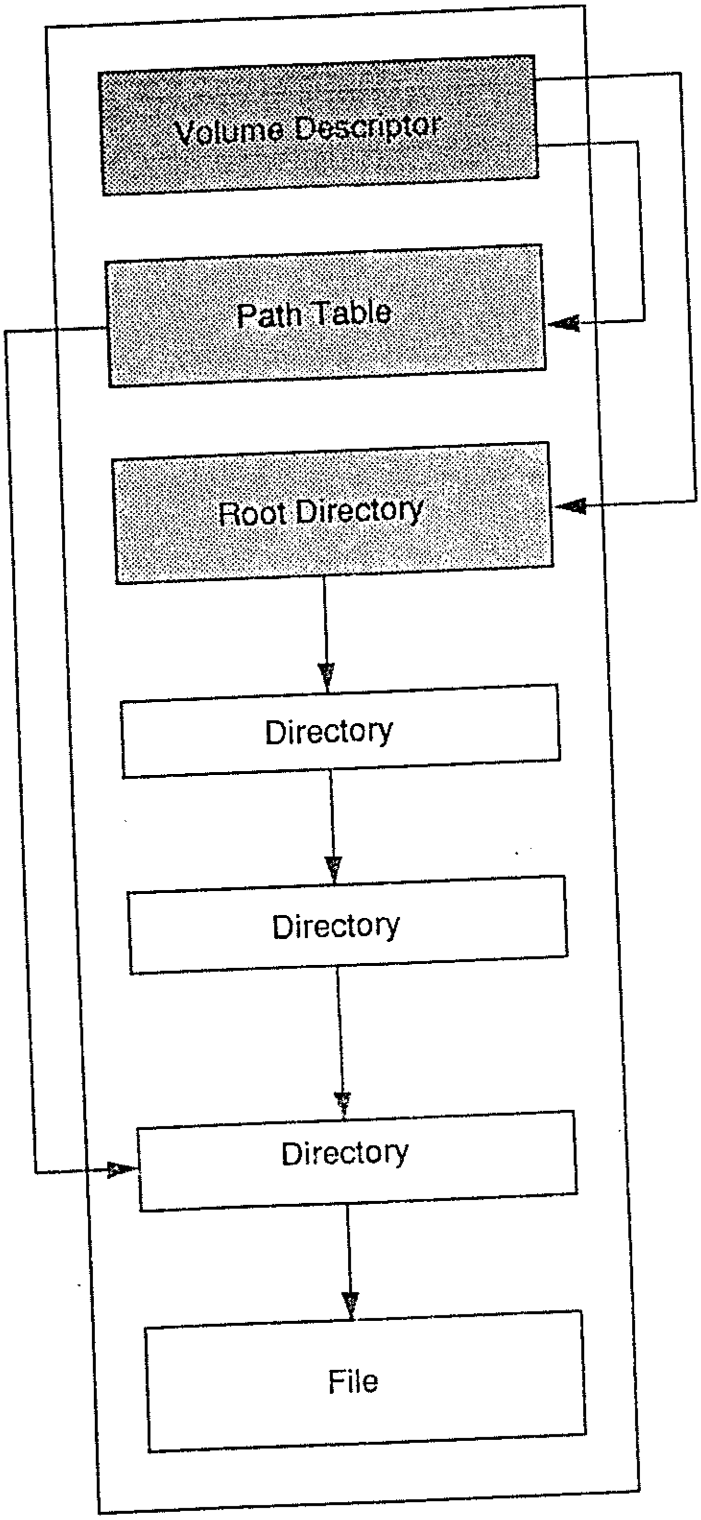
ISO 9660 is designed to accommodate the unique features of CD-ROM, including CD-ROM's large storage capacity, read-only capability, and slow seek time. A hierarchical directory structure is used to organize files efficiently.

The SSU understands and uses the file structure design to access files. In a normal program, however, the programmer does not need to know the details of this file structure design. To use the SSU routine, it is sufficient to be familiar with the MS-DOS or UNIX file structure.

The volume file structure used on CDs will be described in the following section. It differs from MS-DOS in its use of a path table.

3.4.3 THE VOLUME DESCRIPTOR, PATH TABLE AND DIRECTORY

The figure below depicts how files are accessed.



Volume Descriptor
This area on the disc is used to store basic information about the disc, such as the production date. It also gives the location of the path table and root directory.

Path Table
This table indicates the location and parent directory of each directory. Once the path table is loaded into RAM, it may be referenced to find the directory for a particular file. The path table is designed to minimize the data access frequency.

To improve file access time, the path should be loaded into RAM. Then the SSU can use the path table to access files.

Directory
A directory contains the sector numbers for a file. In a hierarchical directory structure, access time is excessive if a search is conducted beginning at the root directory and proceeding through each successive layer.

If the path table is not loaded into RAM, the SSU must follow the hierarchical structure to access a file.