TWAIN

A standard interface was designed to allow application to interface with different types of input devices such as scanners, digital still cameras, digital still cameras, and so on, using a generic TWAIN interface without creating a device- specific driver. The benefits of this approach are as follows

I. Application developers can code to a single TWAIN specification that allows application to interface to all TWAIN complaint input devices.

2. Device manufactures can write device drivers for their proprietary devices and, by complying to the TWAIN specification , allow the devices to be used by all TWAIN-compliant applications

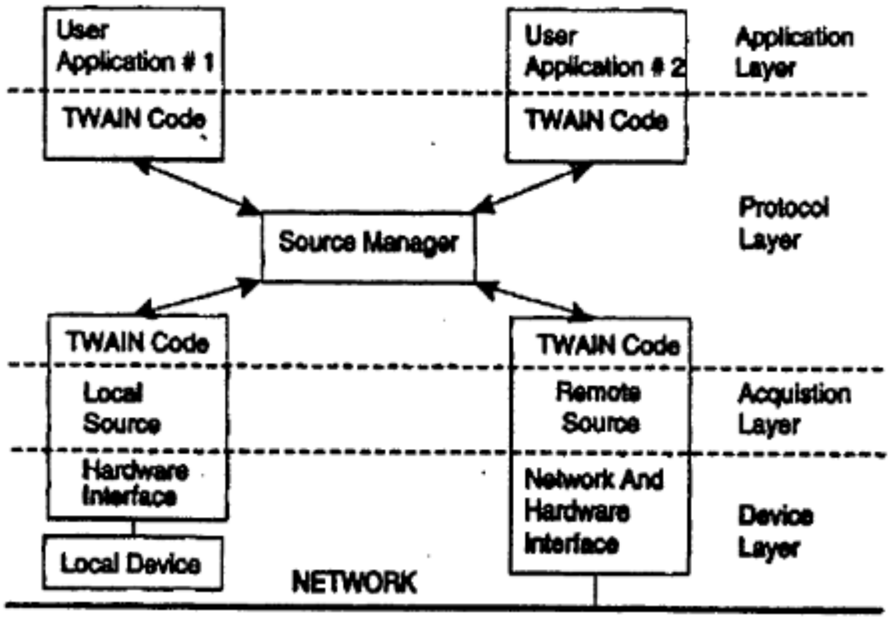
TWAIN Specification Objectives

The TWAIN specification was started with a number of objectives:

- Supports multiple platforms: including Microsoft Windows, Apple Macintosh OS System 6.x or 7.x, UNIX, and IBM OSl2.
- Supports multiple devices: including scanners, digital camera, frame grabbers etc.
- Standard extendibility and backward compatibility: The TWAIN architecture is extensible for new types of devices and new device functionality. New versions of the specification are backward compatible.
- Easy to use: The standard is well documented and easy to use.

The TWAIN architecture defines a set of application programming interfaces (APls) and a protocol to acquire data from input devices.

It is a layered architecture consisting of a protocol layer and an acquisition layer sandwiched between the application and device layers. The protocol layer is responsible for communication between the application and acquisition layers. The acquisition layer contains the virtual device driver to control the device. This virtual layer is also called the source.



TWAIN Architecture

Application Layer: A TWAIN application sets up a logical connection with a device. TWAIN does not impose any rules on the design of an application. However, it set guidelines for the user interface to select sources (logical device) from a given list of logical devices and also specifies user interface guidelines to acquire data from the selected sources.

The Protocol Layer: The application layer interfaces with the protocol layer. The protocol layer is responsible for communications between the application and acquisition layers. The protocol layer does not specify the method of implementation of sources, physical connection to devices, control of devices, and other device-related functionality. This clearly highlights that applications are independent of sources. The heart of the protocol layer, as shown in Figure 2.9, is the Source Manager. It manages all sessions between an application and the sources, and monitors data acquisition transactions.

The functionality of the Source Manager is as follows:

- Provide a standard API for all TWAIN compliant sources
- Provide selection of sources for a user from within an application
- Establish logical sessions between applications and sources, and also manage sessions between multiple applications and multiple sources
- Act as a traffic cop to make sure that transactions and communication are routed to appropriate sources, and also validate all transactions
- Keep track of sessions and unique session identities
- Load or unload sources as demanded by an application
- Pass all return code from the source to the application
- Maintain a default source

The Acquisition Layer: The acquisition layer contains the virtual device driver, it interacts directly with the device driver. This virtual layer is also called the source. The source can be local and logically connected to a local device, or remote and logically connected to a remote device (i.e., a device over the network). The source performs the following functions:

~ Control of the device.
~ Acquisition of data from the device.
~ Transfer of data in agreed (negotiated) format. This can be transferred in native format or another filtered format.
~ Provision of a user interface to control the device.

The Device Layer: The purpose of the device driver is to receive software commands and control the device hardware accordingly. This is generally developed by the device manufacturer and shipped with the device.

OPTICAL MEDIA

Optical drives - CD-ROM, WORM (write once, read many), and rewriteable optical systems - are very appealing for storing large volumes of data. By placing a number of disks in an optical disk library (jukebox), they can significantly increase the storage at almost on-line levels. By and large optical media is fairly indestructible and unaffected by magnetic fields or water. Optical disks provide multi• gigabyte storage for backup and archival, and requirements for large-volume storage.

Optical media can be classified by technology as follows:

- ~ CD - ROM - Compact Disc Read only memory
- ~ WORM - Write Once Read many
- ~ Rewriteable - Erasable
- ~ Multifunction - WORM and Erasable

WORM (Write Once Read Many) optical disk technology records data using a high power laser to create a permanent burnt - in record of data. The laser beam makes permanent impressions on the surface of the disk by creating pits so that once the information is written, it cannot be written over and cannot be erased. Hence the name Write Once Read Many. Once the disk is full, it becomes a read only disk. There are many applications, such as legal, stock trading management and medical, where security of writing data once is of atmost importance. It provides a level of security whereby data cannot be mistakenly erased or altered.

The optical disk consists of six layers. The first layer is a polycarbonate substrate. The next three layers are multiple recording layers made from antimony - selenide (Sb2Se3) and bismuth - tellurium (Bi2Te3). Bismuth - tellurium is sandwiched between antimony - selenide as shown in figure 4.5. The recording layers are covered by aluminium alloy or gold to increase the reflectivity of the recorded surface. The reflective surface is protected by coat of lacquer to prevent oxidation.

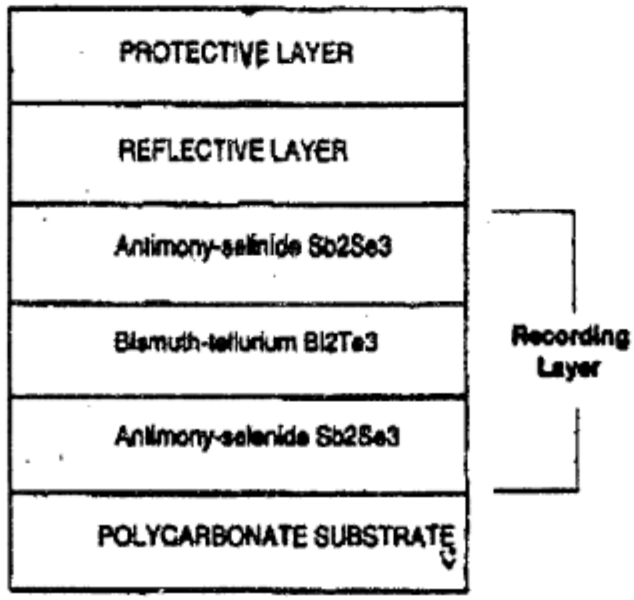


Figure 4.5 : Layers of Worm Drives

During recording, the input signal is fed to a laser diode. The laser beam from the laser diode is modulated by the input signal, which switches the laser beam on and off. When the laser beam is switched on, it strikes the three recording layers. The laser beam is absorbed by the bismuth-tellurium layer (Bi2Te3) and heat is generated within the layer. The heat diffuses the atoms in three recording layers, which result in the formation of four-element alloy (Sb, Se, Bi, Te) layers which is now the recorded area. Figure 4.6 shows the beam-splitting arrangement for a non- reflective higher power laser beam.

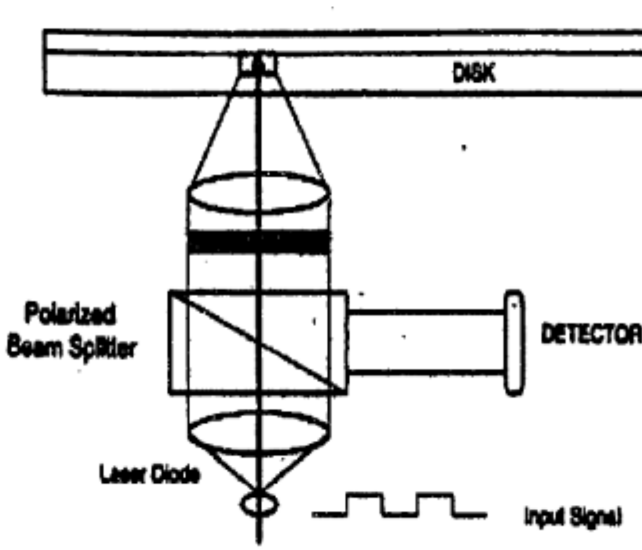


Figure 4.6: Beam Splitting on WORMs for Disk Writes

During disk reads, a weaker laser beam than the write laser beam is focused on the disk. The laser beam is not absorbed due to the reduced power level. Instead it is reflected back. The beam splitter mirror and lens arrangement sends the reflected beam to the photo detector. The photo sensor detects the beam and converts it into electrical signal. Figure 4.7 illustrate the beam splitting action for a lower- powered reflected beam.

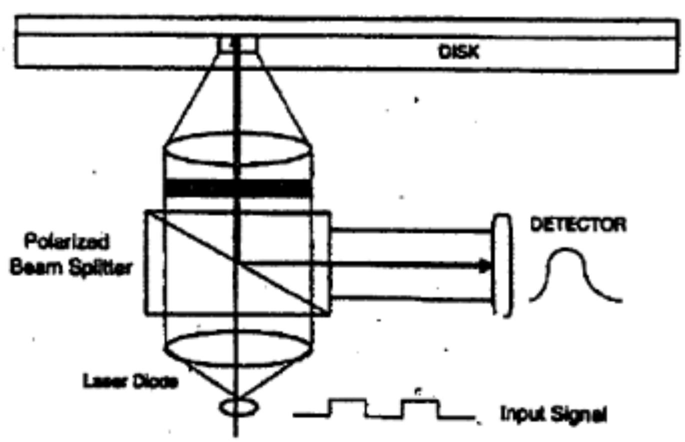


Figure 4.7: Beam-splitting on WORMs for Disk Reads

WORM Format Standards:

The 5-1/4" form factor has become the standard in the industry. The trend is towards the even smaller 3 - 1/2" form factor.

WORM Performance:

Average seek time in a WORM drive is between 70-120 ms as compared to average seek times of 10-25 ms for PC - class magnetic drives. Average latency for optical disks is about 40 ms. and typical data transfer rates fall in at around 800 Kfsytes/sec, although burst transfer rates can be as high as 5 MBytes/sec.

WORM Drive Application:

The Following lists some application that are excellent candidates for WORM drives:

On-line catalogs such as for an automobile parts dealer

» Large-volume distribution

» Transaction logging such as for a stock trading company

» Multimedia archival

Joint Photographic Experts Group Compression (JPEG)

The Joint Photographic Experts Group (JPEG), formed as a joint ISO and CCITT working committee, is focused exclusively on still image compression. Another joint committee, known as the Motion Picture Experts Group (MPEG), is concerned with full-motion video standards. The CCITT Group 3 and Group 4 standards were not designed for gray-scale or color components of an image and are unable to compress image sufficiently for viable operations. The broad JPEG standards were designed to address price performance of a variety of applications using high-resolution graphics. JPEG is a compression standard for still color images and gray - scale images, otherwise known as continuous-tone images.

Requirements Addressed by JPEG:

The JPEG standard was designed to provide a common methodology for compression of continuous-tone images, i.e: for images not restricted to dual-tone (black and white). The following lists some key guidelines for the standardization team:

~ The design should address image quality in the range where visual fidelity is very high and an encoder can be parameterized to allow the user to set the compression or the quality level.

~ The compression standard should be applicable to practically any kind of continuous tone digital source image and should not be restricted by dimensions, color aspect ratios, class, imagery or scene content, or range of shades or colors.

~ It should be scalable from completely lossless to lossy ranges to adapt it to varying storage, CPU, and display requirements.

~ It should provide for progressive encoding. With progressive encoding, the image is decomposed so that a coarser image is displayed first and is filled in as more components of the image are decomposed to provide a finer version of the image.

~ The compression standard should provide the *lossless encoding* .so that images, if needed, can be guaranteed to provide full detail at the selected resolution when decompressed.

Overview of JPEG Components:

The basic component of the standard is the *baseline sequential codec.* Even the baseline sequential codec is a rich and sophisticated compression method adequate for most applications. JPEG standard components are:

);> Baseline sequential codec

);> DCT progressive mode

);> Predictive lossless encoding

);> Hierarchical mode

Baseline Sequential Codec: The baseline sequential codec consists of three steps: formation of OCT coefficients, quantization, and entropy encoding. The baseline sequential codec is a rich compression scheme sufficient for most applications. Huffman coding is used for entropy encoding.

DCT Progressive Mode: The key steps of formation of OCT coefficients and quantization are the same as for the baseline sequential codec. The key difference is that each image component is coded in multiple scans instead of a single scan. Each successive scan refine the image until the picture quality established by the quantization tables is reached.

Predictive lossless Encoding: This was set up as a simple predictive method, independent of OCT processing, to define a means of approaching lossless continuous- tone compression. A predictor combines sample areas and predicts neighboring areas on the basis of the sample areas. The predicted areas are checked against the fully lossless sample for each area, and the difference is encoded losslessly using man or arithmetic entropy encoding.

Hierarchical mode: The hierarchical mode provides a means of carrying multiple resolutions. Each successive encoding of the image is reduced by a factor of two, in either the horizontal or vertical dimension. This is useful if a very-high-resolution image must be accessed by a lower-resolution device which does not have the buffer space for a high-resolution reconstruction of the image. This carries the lowest resolution supported and enough differential information for image resolution in steps of multiples of two for decoding back to the full resolution.

JPEG

Methodology:

The JPEG compression scheme is lossy, and utilizes forward discrete cosine transform, a uniform quantizer, and entropy encoding. The DCT function removes data redundancy by transforming data from a spatial domain to a frequency domain; the quantizer quantizes DCT coefficients with weighting functions to generate quantized DCT coefficients optimized for the human eye; and the entropy encoder minimizes the entropy of the quantized DCT coefficients .By this methodology ,the reduction of a large volume of data to a smaller version is achieved, discarding information that has little visual effect, and further compression of the data by taking advantage of its spatial characteristics.

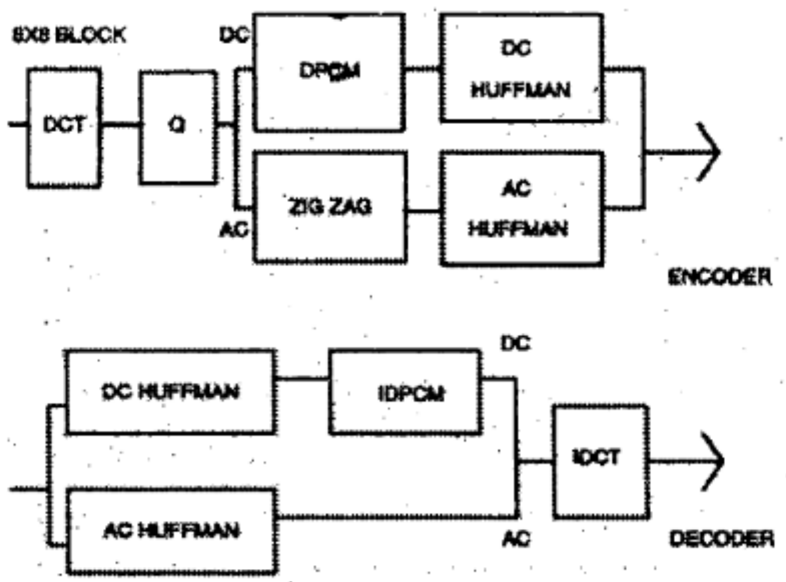


Figure 2.1: Symmetric Operation of DCT Based Codec.

The Discrete Cosine Transform (DCT)

DCT is a mathematical operation closely related to Fourier transform. In the time domain , the signal requires lots of data points to represent the time in x-axis and the amplitude in y-axis. Once the signal is converted to a frequency domain using Fourier transforms, only a few data points are required to represent the same signal, because signal contains only a few frequency components. This allows us to represent the signal with only a few data points in a frequency domain that would take lots of data points if represented in a time domain. This technique can be applied to a color image. A color image is composed of pixels.

These pixels have RGB color values, each with its x and y coordinates using an 8 x 8 or 16 x 16 matrix for each primary color. When considered over an 8 x 8 matrix of 64 values, each with x and y coordinates, we have a three dimensional representation of pixels called a spatial representation or spatial domain. The spatial representation is converted to a spectral representation or frequency domain by OCT conversion.

DCT
Coefficients

Each 8 x 8 block of source image sample is effectively a 64-point discrete signal which is a function of two spatial dimension x and y. If this signal is decomposed into 64 orthogonal basis signals, each of these 64 signals will contain one of the 64 unique two dimensional spatial frequencies which make up the input signal's spectrum. The output amplitudes of the set of 64 orthogonal basis signals are called DCT Coefficients. In other words, the value of each OCT Coefficient is uniquely defined by the particular 64- point input signal. The coefficient with zero frequency in both dimensions is called the DC Coefficients and the remaining ones are called AC coefficients.

Quantization

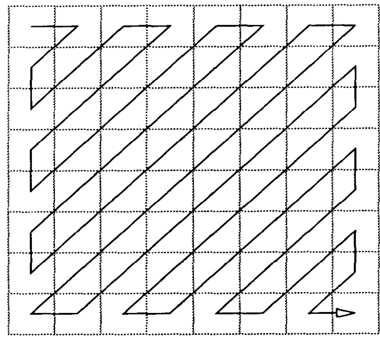
This is the main lossy operation in the whole process.

After the DCT has been performed on the 8x8 image block, the results are quantized in order to achieve large gains in compression ratio. Quantization refers to the process of representing the actual coefficient values as one of a set of predetermined allowable values, so that the overall data can be encoded in fewer bits (because the allowable values are a small fraction of all possible values).

After quantization, most quantized DCT coefficients in the 8 x 8 matrix (fractional values) are truncated to zero values. Therefore, the matrix ends up with most OCT values as zeroes and some nonzero DCT values. JPEG has elected to compress zero values by utilizing the run-length scheme. Run-length encoding is a code to represent the count of zero-value DCT coefficients. This process of run-length encoding gives an excellent compression of the block consisting mostly of zero values.

Zigzag Sequence

Further empirical work proved that the length of zero values in a run can be increased to give a further increase in compression by reordering the runs. JPEG came up with ordering the quantized DCT coefficients in a zigzag sequence



Entropy Encoding

Entropy is defined as a measure of randomness, disorder, or chaos, as well as a measure of a system's ability to undergo spontaneous change. The entropy encoder compresses quantized DCT coefficients more compactly based on their spatial characteristics. The baseline sequential codec uses Huffman coding.It is mathematically represented as follows:

Entropy in number of bits = log2(probability of object)

In this equation, the object can be a character.