# Designing DVD player trick plays

**Trick plays provide the hallmark of an advanced DVD player. In your next DVD player design, it is important to consider the efficient interaction of your local interface drive, decoder circuitry, and navigation software.**

**By Alain Bismuth**
Director, Consumer DVD Products
LSI Logic

**By Gordon Ng**
DVD Engineering Manager
LSI Logic

You will find new design challenges associated with next-generation DVD players. These advanced designs demand tight and efficient interaction among the DVD local interface drive, decoding circuitry, and the system control software. This high level of efficient interaction is vital for you to achieve the high-quality trick plays consumers expect from any digital audio/video equipment. Trick plays include functions such as fast/slow-forward, fast/slow-reverse, pause, viewing-angle changes, and zoom.

DVD players use advanced control software to fully utilize the powerful features present in the video disks available today. This software includes both a presentation engine and a navigation engine. The presentation engine uses information in the presentation data stream from the disk to control content. The navigation engine uses the information in the navigation data stream from the disk to provide a user interface, create menus, control branching, and also to direct the trick plays.

## Dedicated vs. ATAPI drives

When starting your DVD player design, it is helpful to understand the differences between various drives when you begin to evaluate alternatives. In some designs, using a conventional ATA packet interface (ATAPI) disk drive can limit the performance of a DVD player's navigation software and decoding circuitry. Since it has only a limited embedded intelligence, an ATAPI drive puts a major burden on the system microprocessor to control it.

You should be aware that the ATAPI protocol lacks critical, intelligent features like track buffering and NAVI packs. This means it is essentially leveraging extremely high CPU bandwidth available in a PC environment to perform large amounts of data fetching. In an embedded consumer electronics system like a DVD player, this overburdens the small-system CPU with unnecessary processing, which degrades the quality of your crucial trick-play design. This is not the case with the more intelligent local-interface DVD drives, which are better suited for DVD player applications.

## Seamless transitions

You will encounter a new set of terminology as you get into the portion of your design that deals with navigation software routines. Let us consider some of the more important ones. The program stream contains five packetized elementary streams (PES): video, audio, sub-picture, presentation control information (PCI), and data search information (DSI).

DSI is the navigation data used for searching and executing the seamless playback of a video object unit (VOBU). DSI consists of five segments:

- DSI general information,
- Seamless playback information,

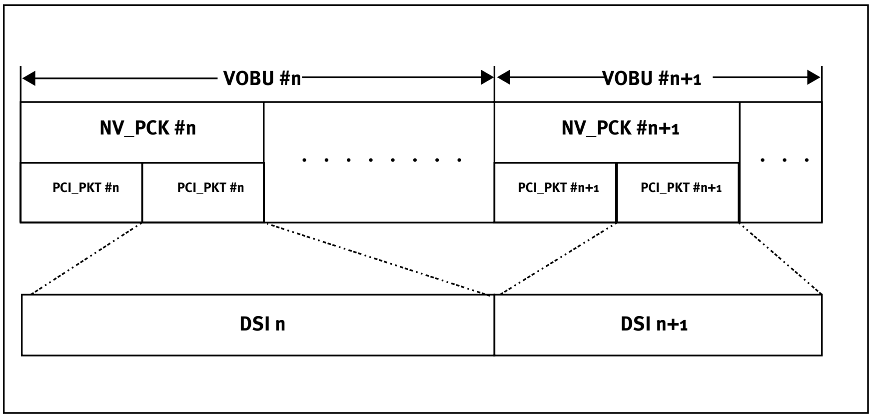


**Figure 1**: DSI is described in its packet (DSI_PKT), which is in the navigation pack (NV_PCK). Its contents are renewed for every VOBU.

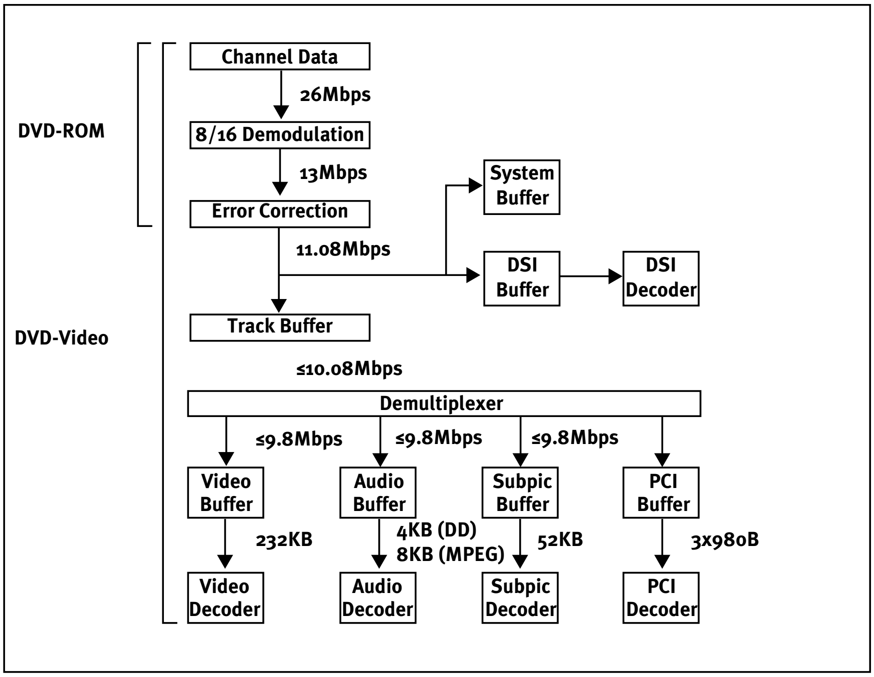


**Figure 2**: DSI, which occupies 1,017 bytes, is copied from the data stream before it reaches the track buffer.

- Angle information for seamless,
- VOBU search information, and
- Synchronous information.

Including the reserved space, you need a total of 1,017 bytes (**table 1**). DSI is described in its packet (DSI_PKT), which is in the navigation pack (NV_PCK). Its contents are renewed for every VOBU (**figure 1**).

DSI is copied from the data stream before it reaches the track buffer (**figure 2**). The track buffer is a FIFO used for stopping underflows during seamless trick plays. It provides a variable stream of data up to 10.8Mbps to the system decoders of data coming from the disk at 11.08Mbps.

In a segment where seamless playback of multiple paths is carried out, the layout of the presentation data has an interleaving structure (**figure 3**). In an interleaved block, the presentation engine follows the specified playback path to play the presentation data, while reading data sequentially and skipping data that it does not need.

While jumping, the presentation engine controls the track buffer to prevent an intermittent supply of presentation data to the decoder. During the jump, you can guarantee a continuous data supply to the decoder by controlling the data quantity in the track buffer. How? By using the difference between data transfer rate from the disk to the track buffer (Vr), the consumption rate at the decoder (Vo), and the layout of data on the disk.

You will find that a key component of high-quality DVD trick plays is an efficient DSI scheme. Here, the DVD A/V decoder processes and interprets the DSI in parallel before the data stream passes through the track buffer. Therefore, the execution of trick plays is anticipated, resulting in extremely smooth, seamless transitions.

## Data overlays

Presentation data structure is overlaid on the physical data structure with top level for titles. Each title contains up to 999 program chains (PGCs). A PGC contains 0 to 99 programs, which are ordered collections of pointers to cells. Physical data and the logical presentation data structure converge at the cell level. The PGC connects cells together, defines the order programs, and which cells are played.

Programs within a PGC are used for sequential, random, or shuffle play, and cells can be used by more than one program chain. Groupings include parental management blocks, angle blocks for multiple camera angles, interleaved blocks for seamless branching, etc.

Navigation data structure is overlaid on the physical data structure. It is in four categories: control, search, user interface, and navigation commands. Of particular interest is search data, which creates a navigational structure.

Control information describes the navigation data for NTSC or PAL formats, aspect ratio, language, audio and sub-picture selection, plus

| Section | Content | Number of bytes |
|---|---|---|
| DSI_GI | DSI general information | 32 |
| SML_PBI | Seamless playback information | 148 |
| SML_AGLI | Angle information for seamless | 54 |
| VOBU_SRI | VOB unit search information | 168 |
| SYNCI | Synchronous information | 144 |
| Reserved | Reserved | 471 |
| | Total | 1,017 |

**Table 1**: Data search information (DSI) is the navigation data used for searching and executing the seamless playback of a video object unit (VOBU).

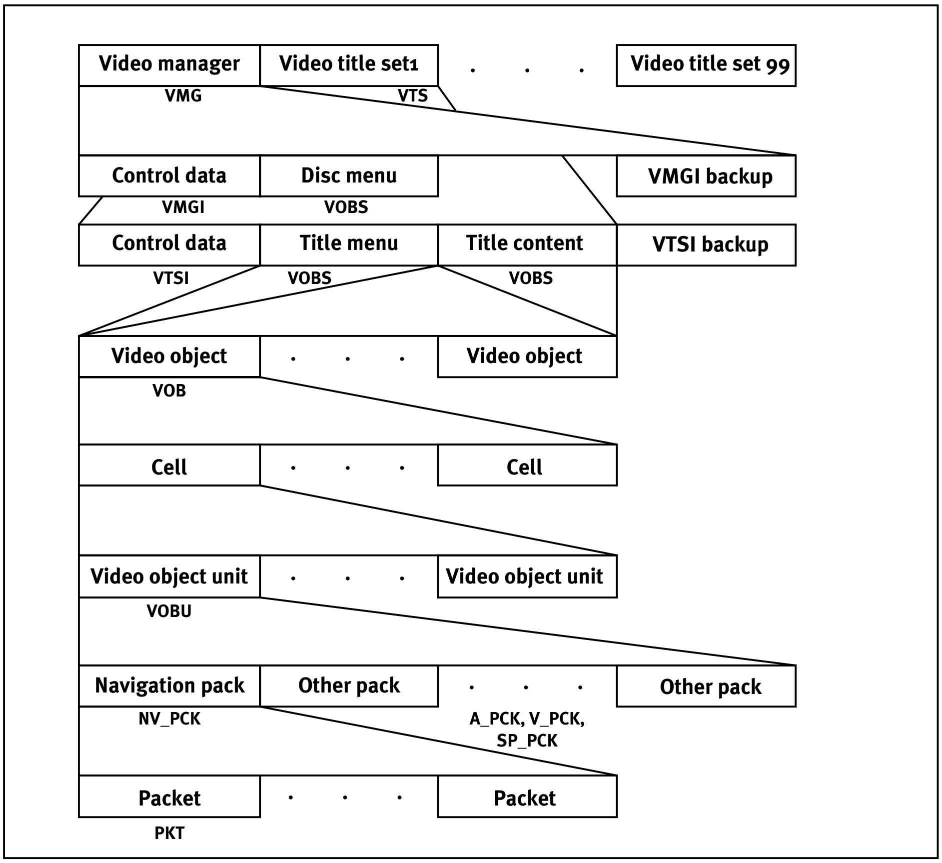


**Figure 3**: In an interleaved block, the presentation engine follows the specified playback path to play the presentation data, while reading data sequentially and skipping unnecessary data.

moral codes for parental control. Navigation commands, on the other hand, provide branching and other interactive features.

Navigation commands are like CPU instruction words and are instructions to the DVD player's microprocessor. Each command is up to 8 bytes long, and can be one, two, or three instructions. Instructions include math operations, logical operations, comparison, register operations, and flow control.

## Design issues

For you to properly engineer efficient angle changes, it is important to include an optimized DVD decoder with a specialized DVD drive. Angle changes are performed at the cell level. Data from nine separate angles, which are simultaneous in time, are stored in the angle block on the DVD disk.

Information for each angle change is part of the DSI. Ideally, each angle change should be performed with a smooth transition and no video breakup or jerks. You will find this to be very difficult because it means the drive's head must move to another disk location to play that information. This task must be done quickly to eliminate underflow of the data buffers in the A/V decoder, thus avoiding errors.

## Seamless angle change

The bit stream for an angle is stored in a structure referred to as an angle block. The angle block is constructed from a multiplex of VOBUs called interleaved VOBUs (ILVU). Each ILVU for a different angle number is simultaneous in time. Therefore, if your design allowed every ILVU in an angle to play, the viewer will see a jump backward in time as one angle finishes and another starts. To stop this problem, the drive needs to seek the next ILVU while in the current angle.

Assuming a worst-case search of 250ms (about eight video frames), this delay is much larger than the video buffer verifier (VBV) buffer defined by MPEG. During this time, the channel buffers in the MPEG decoder

| SML_AGLI | Content | Number of bytes |
|---|---|---|
| SML_AGL_C1_DSTA | Address and size of destination ILVU in AGL_C1 | 6 |
| SML_AGL_C2_DSTA | Address and size of destination ILVU in AGL_C2 | 6 |
| SML_AGL_C3_DSTA | Address and size of destination ILVU in AGL_C3 | 6 |
| SML_AGL_C4_DSTA | Address and size of destination ILVU in AGL_C4 | 6 |
| SML_AGL_C5_DSTA | Address and size of destination ILVU in AGL_C5 | 6 |
| SML_AGL_C6_DSTA | Address and size of destination ILVU in AGL_C6 | 6 |
| SML_AGL_C7_DSTA | Address and size of destination ILVU in AGL_C7 | 6 |
| SML_AGL_C8_DSTA | Address and size of destination ILVU in AGL_C8 | 6 |
| SML_AGL_C9_DSTA | Address and size of destination ILVU in AGL_C9 | 6 |
| | Total | 54 |

**Table 2**: If a change of angle is requested, the angle information for the seamless table determines where to read the next ILVU.

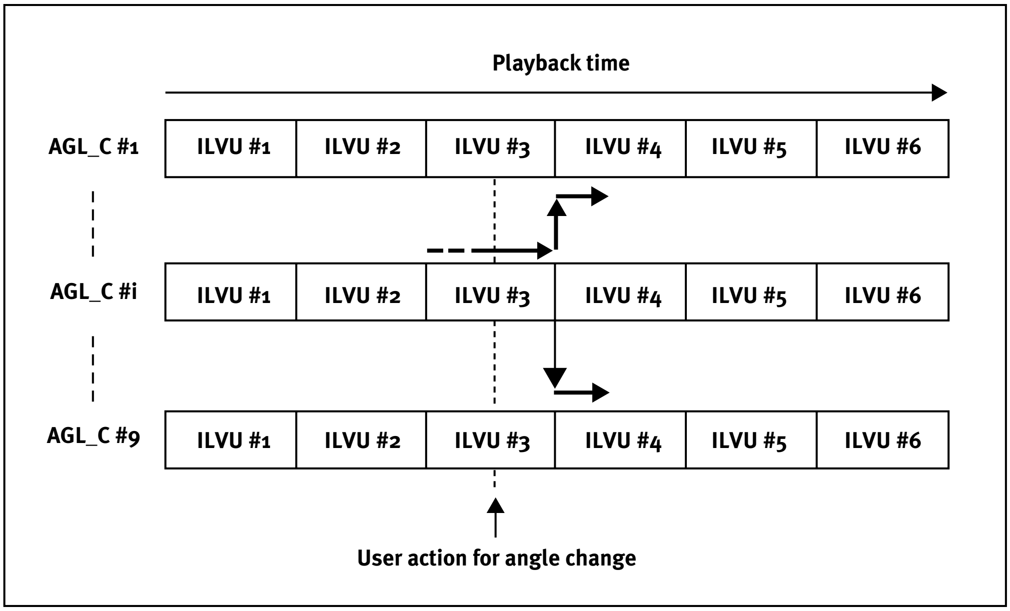


**Figure 4**: During normal playback of an angle block, the DSI information is parsed to find the next ILVU start address. This controls the head to read the next ILVU in the current angle.

will underflow. This creates unsightly artifacts or a playback pause.

To fix this problem, you must use a track buffer FIFO that is large enough to compensate for the disk seek time. In the DVD specification, you will find an example calculation for a track buffer size, which shows it to be about 3.3Mb. In a real system, the size of the track buffer may vary, depending on the drive seek time and transfer rate.

During normal playback of an angle block, the DSI information is parsed to find the next ILVU start address. This controls the head to read the next ILVU in the current angle. If a change of angle is requested, the angle information for the seamless table determines where to read the next ILVU (**figure 4** and **table 2**). A skip on the disk is required for each angle change, and a seek time of no more than 250ms is issued when no data is coming from the drive and going to the decoder. In this case, there is a chance that the decoder might not receive data to decode. Considering that each picture is 30ms and there is 250ms of seek time, this becomes a serious design issue.

The track buffer—a 3Mb or 4Mb FIFO that stores compressed data coming from the disk—minimizes this problem. To implement angle changes efficiently, processing of the DSI information is tightly coupled to the precise moment the data comes off the disk. This minimizes the delay from the point the decision is made to do an angle change to the end of the operation. You must be careful that it takes no more than 250ms because, otherwise, there is a chance of underflowing the track buffer.

Therefore, you need a DVD drive with its own track buffer integrated into that subsystem to minimize delays. Drives like this read DSI as well as data from the disk. When an angle-change command is used, the drive moves to that angle. When an angle change is performed, the DSI information is required before it returns to the track buffer, not after.

The importance of having early access to the DSI occurs when a small-size VOBU is processed right after a large VOBU. If the DSI is not available early, the small VOBU causes the track buffer to underflow. In your designs, you must ensure that sufficient data stream is fed into the track buffer, achieving seamless transition for trick plays.

A specialized DVD drive (from Sanyo, for example) has built-in intelligence that allows it to perform efficient angle changes. This type of drive supports built-in track buffering and provides DSI earlier to the A/V decoder for look-ahead processing. Other intelligence includes the ability to understand a set of high-level macro commands specific to a DVD application. The intent is to minimize the burden on the A/V decoder or the system CPU, allowing more advanced trick-play processing.

### Faster fast-forward/reverse

In some current DVD players, the video picture is distorted when placed in these modes. **Figure 5** shows the composition of a VOBU,

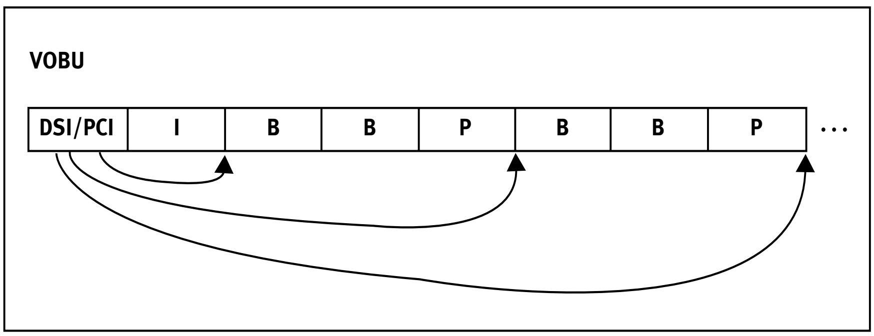


**Figure 5**: DSI and PCI are at the start of each VOBU, which is the lowest level of navigation software hierarchy.

which is the lowest level of navigation software hierarchy. DSI and PCI are at the start of each VOBU.

In MPEG terminology, reference frames are frames other than B frames. To decode a B frame, the I and P frames on either side of the B frames are needed before the frames in the middle are decoded. Decoding B frames consumes processing power. To design an efficient fast-forward mode, B frames are omitted and only certain I and P reference frames are played.

The term "fast forward/reverse" is a misnomer because the assumption is that the drive system is able to operate faster than normal. Most drives can do 2x faster playback, not sufficient for smooth DVD trick plays. At times, trick plays demand 8x to 15x faster playback capabilities. This means playing fewer VOBUs so more information is processed faster. The DSI information is used to control the drive so that only the first three reference frames of a VOBU are played. The optimized decoder removes the interleaved B frames so only the "I's" and "P's" are deployed, resulting in a powerful fast-forward mode.

Specialized DVD drives like Sanyo's are important contributors to high-quality trick plays. Some DVD player designs use the conventional ATAPI drive, but ATAPI is designed for very short, very fast accesses necessary in PC and not optimized for constantly playing movies. The combination of a good drive with a high-performance DVD decoder, as also optimized control software, is the key to the building your next-generation DVD players. ee

**You may send your comments on this article to Alain Bismuth through e-mail at bismuth@lsil.com or Gordon Ng at ng@lsil.com, fax: 1-408-9544451.**