3.3 Level II Programs Encoded on Videodiscs

As mentioned at the beginning of this chapter, programmers preparing a Level II program to be encoded onto a videodisc will most likely use an authoring utility program of some sort. Programmers might develop their own compilers and other utilities or use commercially available authoring support tools. You may contact Pioneer Communications of America, Inc., Multimedia Engineering/Technical Support at (201) 327-6400 for more information.

It is highly recommended that the entire Level II application be carefully structured before any programming begins. Like scriptwriting and storyboarding, the interactive program should be well thought out and reasonably well documented before any video or audio production begins. This will help eliminate the need for the costly re-do of video sequences that don't quite fit into the intended interactive framework.

A flow chart provides a symbolic roadmap for any interactive application. The intent of the flowchart is to clearly document the intended interactions and sequence of visual and audio events. The specific symbols used are not too important. Consistency and clarity are very important. For your convenience, we have provided example flow chart symbols in ***Appendix F, Sample Flow Chart and Level II Code***. Additional information on flow charts is also available from Multimedia Engineering/Technical Support.

The actual Level II program code is encoded on Audio Channel 2 of the videodisc as a brief series of audio tones. Each "burst" of tones lasts about 2 seconds and contains one page of data (1022 bytes). It is sometimes referred to as a Level II program "Dump".

Caution: When planning to place program dumps one after another on the videodisc, remember to allow for sufficient space between the dumps. PVMI specifies 3 seconds (90 frames) per dump, allowing for 0.5 seconds of leader tone (30 fields), almost 2 seconds for the dump itself (approximately 100 fields) and about another 0.5 second of buffer zone. Refer to **Figure 3-C**.

Note: The vertical line below in the leader represents the target frame for the dump.

Structure of a Pioneer Level II "Dump"

Leader tone.	Level II Program Code	Buffer Zone
0.5 sec.	1.64 sec.	0.5 sec.
15 Frames	50 Frames	15 Frames

90 frames on Audio Channel 2

Figure 3-C

When a Level II videodisc is spun up on the LD-V8000 and the ***Level II Auto Start*** *On-Screen Function Switch Setting* is set to ***Load from Disc***, the player searches to frame 1, squelches the video and audio, and looks for Level II leader tone on the disc's Audio Channel 2.

When the player finds leader tone, it loads the first dump into memory page one and then begins executing the Level II program from program address zero. As the program in that first page of memory is being executed, the program can command the player to search to any other frame on the videodisc and attempt to load another page of program code (another dump). If a dump is found, it will be read and either overlay previously loaded memory or fill unused memory, as commanded by the Level II program instructions.

If the player finds no Level II leader tone at frame 1, it continues into the player's normal *Manual Mode*, to be controlled by either Level I or Level III commands. Later, one of those commands may initiate Level II operations, executing any Level II program codes that might be in the player's memory.

Preparing Level II Programs for Transmission to PVMI

When a Level II videodisc is manufactured by Pioneer Video Manufacturing, Inc. (PVMI), each Level II program dump is sent to PVMI as an "Object File" in Pioneer Level II Object Code Format. Along with comments, the target frame number, and possibly other information, the file contains a series of Hex codes, one two-character Hex code for each byte in the program dump. The file also contains a check sum value of object code at the end of the dump.

The Object File is a simple ASCII text file. Contact Pioneer Multimedia Engineering/ Technical Support for a document describing Pioneer's Level II Object File syntax.