

LIB File Format

By unknown

Document ID: Q79259

Product: Microsoft BASIC Compiler

Title: Microsoft Library (.LIB) Format, Created by LIB.EXE

Updated: 27-DEC-1991

Operating System Versions: 6.00 6.00B 7.00 7.10

Operating Systems: MS-DOS

Summary:

This article describes the components of the Microsoft Library Format (for .LIB files created by the LIB.EXE Library Manager). The Microsoft Library Format is consistent between compatible Microsoft languages. While future library utilities will remain backward-compatible with the older library formats, the actual library format itself is subject to change. This information is taken from Chapter 5 of the "Microsoft C Developer's Toolkit Reference," which contains more in-depth information on the Microsoft Library Format.

This information applies to Microsoft QuickBasic versions 4.0, 4.0b, and 4.5 for MS-DOS, to Microsoft Basic Compiler versions 6.0 and 6.0b for MS-DOS, and to Microsoft Basic Professional Development System (PDS) versions 7.0 and 7.1 for MS-DOS.

More Information:

Library Header Record

Object code library .LIB files under MS-DOS always contain blocks of data in multiples of 512 bytes. The first record in the library is a library header. This record is structured the same as a Microsoft object-module-format (MS OMF) record. That is, the first byte of the record identifies the record's type, and the next two bytes specify the number of bytes remaining in the record. Note that the length field is byte-swapped (in other words, the low-order byte precedes the high-order byte). The record type for this library header is F0 hex (240 decimal).

Modules in a library always start at the beginning of a page. Page size is determined by adding three (one for the record type byte and two for the record length field itself) to the value in the record length field; thus the library header record always occupies exactly one page. Legal values for page size are given by the range of 2 through the n, where n is a value from 4 through 15.

The four bytes immediately following the length field are a byte-swapped long integer specifying the byte offset within the library of the first block of the dictionary. The next two bytes are a byte-swapped word field that specifies the number of blocks in the dictionary. (Note: The Library Manager, LIB.EXE for MS-DOS, cannot create a library whose dictionary requires more than 251 512-byte pages.)

The next byte contains flags describing the library. One current flag definition is "0x01 = case sensitive". This applies to both regular and extended dictionaries. All other values are reserved for future use and should be 0. The remaining bytes in the library header record are not significant. This record deviates from the typical Microsoft OMF record in that the last byte is not used as a checksum on the rest of the record.

Object Modules

LIB File Format

By unknown

The first object module in the library immediately follows the header. The first object module is followed in turn by all other object modules in the library. Each module is in Microsoft OMF. Individual modules are aligned so that they start at the beginning of a new page. If, as is commonly the case, a module does not occupy a number of bytes that is exactly a multiple of the page size, then its last block is padded with as many null bytes as are required to fill it. This special format is covered in detail in the "C Developer's Toolkit Reference."

Dictionary Blocks

The remaining blocks in the library compose the dictionary. The number of blocks in the dictionary is given in the library header. Dictionary length is in 512-byte blocks. Detailed information on the exact content and format of the dictionary are contained in the "C Developer's Toolkit Reference."

Extended Dictionary

The extended dictionary is optional and indicates dependencies between modules in the library. Versions of LIB.EXE earlier than version 3.09 do not create an extended dictionary. The extended dictionary is placed at the end of the library. Again, see the "C Developer's Toolkit Reference" for details on the structure of the Extended Dictionary.

Additional reference words: 6.00 6.00b 7.00 7.10 4.00 4.00b 4.50