P-Machine Executable File Specification

A P-Machine accepts a file containing a directory, constants, and an array of P-Code instructions as shown in Figure 1.

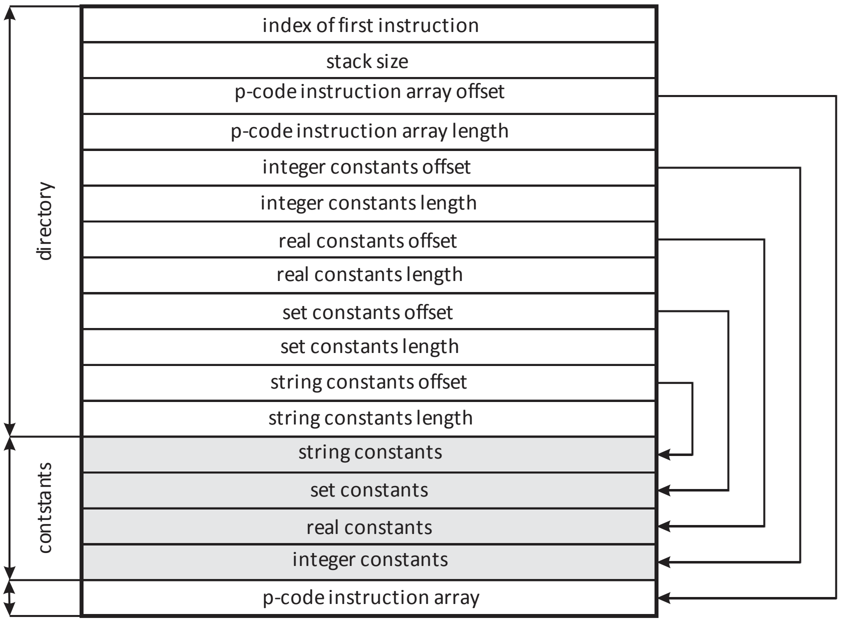

Figure 1. Anatomy of a P-Machine Executable File

The directory is placed at the beginning of the file. Constants are placed after the directory and the P-Code instruction array comes directly after the constants.

1. Directory

The P-Machine program directory has fields containing the offsets to components of the program. Offsets are relative to the start of the file and are given in bytes.

The size of the directory is equal to the size of twelve integers. On an IBM RISC 6000 computer, the size of an integer is four bytes making the size of the directory 48 bytes. On many IBM-compatible personal computers, the size of an integer is two bytes and the size of the directory is 24 bytes.

1.1. Index of first instruction

The index of the first instruction marks the first P-Code instruction to execute. The index of the first instruction designates the beginning of the P-Machine program whose instructions are in the P-Code instruction array.

1.2. Stack size

The stack size is the size of stack allocated for the P-Machine program.

1.3. P-Code instruction array offset

The P-Code instruction array offset is the number of bytes from the beginning of the file to the first byte of the P-Code instruction array.

1.4. P-Code instruction array length

The length of the P-Code instruction array is the number of bytes occupied by that array.

1.5. Integer constants offset

Integer constants are an array of integers. The offset to integer constants is the number of bytes from the beginning of the file to the first byte of integer constants.

1.6. Integer constants length

The length of integer constants is the number of bytes occupied by the array of integer constants.

1.7. Real constants offset

Real constants are an array of real numbers. The offset to real constants is the number of bytes from the beginning of the file to the first byte of real constants.

1.8. Real constants length

The length of real constants is the number of bytes occupied by the array of real numbers.

1.9. Set constants offset

A set is an array of eight bytes. Each bit in the set represents an element of a set. There can be at most 64 elements in a set.

Set constants are an array of sets. The offset to set constants is the number of bytes from the beginning of the file to the first byte of set constants.

1.10. Set constants length

The length of set constants is the number of bytes occupied by the array of sets.

1.11. String constants offset

A string is a sequence of characters terminated by the character whose integer value is zero. P-Machine strings are identical to those defined in the C programming language.

String constants are packed into an array of characters. The offset to string constants is the number of bytes from the beginning of the file to the first character of the first string constant.

1.12. String constants length

The length of string constants is the number of bytes occupied by all the string constants.

2. String constants

String constants are stored in an array. String constants vary in size. P-Machine strings have the same form as C-programming language strings. Strings consist of a sequence of characters terminated by a character whose ordinal value is zero. P-Machine instructions reference string constants by using the index of the first character of the string.

Two strings are shown in Figure 2. The string “candy" is followed by the string "is." Together, both strings require nine (9) characters including their terminators.

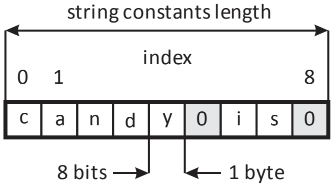

Figure 2. Anatomy of string constants

3. Set constants

Set constants are stored in an array. Each set occupies 64 bits. The number of sets, n_t, is given by n_t=l_t/8, where l_t is the value of the field labeled "set constants length" in the directory.

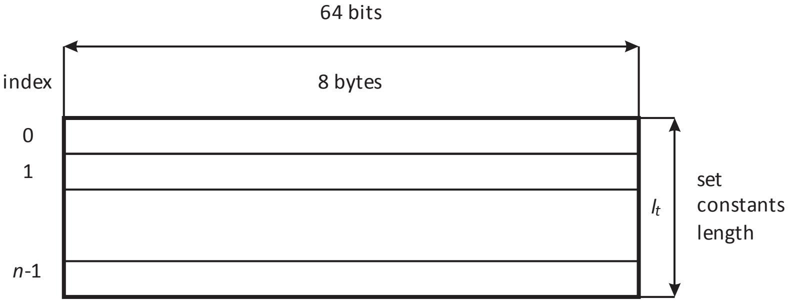

Figure 3. Anatomy of set constants

4. Real constants

Real constants are stored in an array. Each real number occupies 64 bits. The number of real constants, n_r, is given by, n_r=l_r/8, where l_r is the value of the field labeled "real constants length" in directory. Real constants are stored in IEEE 754 binary double format.

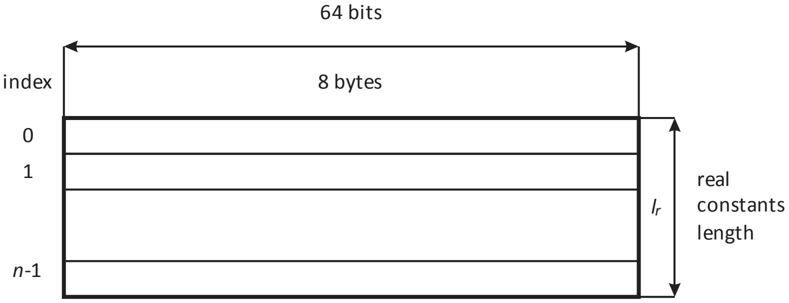

Figure 4. Anatomy of real constants

5. Integer constants

Integer constants are stored in an array. A standard integer occupies four bytes (32 bits) on Intel Computers used by the Department of Computer Science at the University of Central Oklahoma. The number of integers, n_i is given by $n_i = l_i/4$ where l_i is the value stored in the directory in the field labeled "integer constants length."

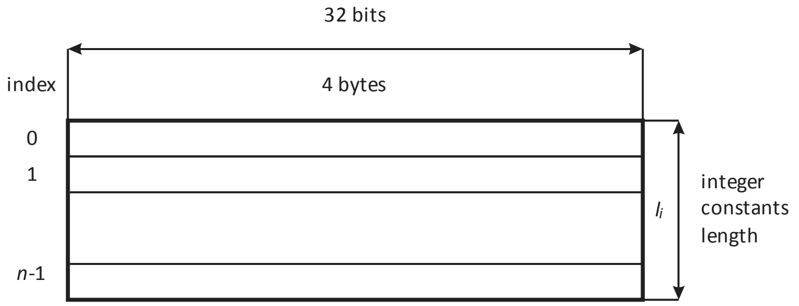

Figure 5. Anatomy of integer constants

6. P-Code instruction array

P-Code instructions are stored in an array. Each P-Code instruction occupies 4 bytes (32 bits). The number of P-Code instructions, n_p, is given by, n_p=l_p/4, where l_p is the value of the field labeled "P-Code instructions array length" in the directory.

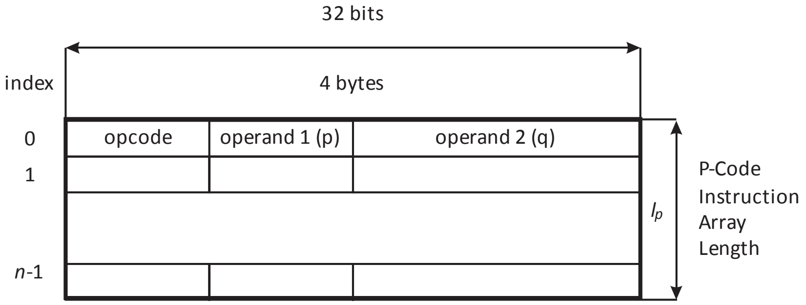

Figure 6. Anatomy of the P-Code instruction array

P-Code instructions consist of three fields, an operation code (opcode) followed by two operands. Operands depend on the opcode.

P-Code instructions occupy 32 bits or 4 bytes. The opcode is stored in the first byte of the instruction. The first operand, called *p*, is stored in the second byte and the second operand, called *q*, occupies the remaining two bytes as shown in Figure 7.

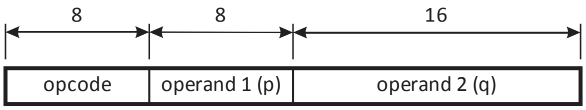

Figure 7. Anatomy of a P-Code instruction

The opcode and both operands are unsigned integers.