



Data is stored in numeric fields in a Line or Record Sequential file as shown below:

Value	PIC	In file
9	999	009
9	s999	009
-9	s999	00y
9	999v99	00900
9.9	999v99	00990
9	s999v99	00900
9.9	s999v99	00990
-9	s999v99	0090p

As shown above, there is no way to tell or determine the decimal position from the value stored in file. If it is known, then the value can be converted easily. For example, the number in file can be divided to 100 to obtain the actual value if the PIC comes with V99.

As shown above, all negative figures end with a letter in file. See the list below to convert the letter to a number:

p	0
q	1
r	2
s	3
t	4
u	5
v	6
w	7
x	8
y	9

If the last character is a letter (between 'p' and 'y'), then it is a negative number. The last character can then be changed to a number using the list above. The rule or logic behind the conversion is to add 64 to the decimal representation of the last digit, and that gives the decimal representation of the last character to take the last position.

Example:

The value in file is 045v (assuming it is a whole number). Since the last position is a letter, the number must be then negative.

According to the conversion table above, the last character should be replaced with 6. Without the conversion table, the following calculation has to be done:

1. determine the decimal representation of 'v', which is 118
2. subtract 118 by 64 -> 54

The character that corresponds to 54 is the number 6. Therefore, 045v is actually -0456 or -456.

Note: The above does not apply to COMP fields. Writing a COBOL program or using an ODBC driver that supports