Über binäre Gleitpunktzahlen – Crashkurs für Stevie ;)

Vorüberlegung:

Normale Kommazahlen im Zehnersystem.

z.B.: 234, 1476

Interpretation vor dem Komma: $4*10^0 + 3*10^1 + 2*10^2$ ... (Einer, Zehner, Hunderter etc. also alles Potenzen zu 10!)

Interpretation nach dem Komma: $1*10^{-1} + 4*10^{-2} + 7*10^{-3} + 6*10^{-4}$
Merke: Die Zählung beginnt nicht mit Einteln, sonder mit Zehntel, dann Hunderstel, ... ==> Negative Potenzen zu 10!
Auch die Reiehnfolge der Ziffern ist umgekehrt. Man zählt von links statt rechts!!!

Wenn das so mit Derzimalzahlen geht, warum sollte es nicht mit Binärzahlen genauso gehen?

Beispiel:
110001,001111

=> $1*2^5 + 1*2^4 + 1* 2^0 + 1*2^{-3} + 1*2^{-4} + 1*2^{-5} + 1*2^{-6}$

Multipliziert man eine dezimale Kommazahl mit 10^x, verschiebt man das Komma um x Stellen nach rechts. Multipliziert man mit 10^{-x}, verschiebt man es um x-Stellen nach links.

Das selbe geht auch mit binären Kommazahlen, nur das die Basis hier nicht 10 ist, sondern 2. Wir sind ja im Zweiersystem.

Was speichert man nun in einer binären Gleitpunktzahl?

- Vorzeichenbit
- Exponent
- Mantisse

Wie berechnet man den Wert einer Gleitpunktzahl?

Vorzeichen * Mantisse * 2^{Exponent}.

Besonderheit: Die Mantisse muss normiert werden. Das heißt, vor dem Komma muss immer eine 1 stehen. Also 1,....!!!!! Weil man auch immer davon ausgeht, dass die Mantisse normiert wird, wird die 1 nicht abgespeichert. => Die Berechnung muss heißen:

Vorzeichen * (1+Mantisse) * 2^{Exponent}!!

<u>Alle Bits der Mantisse müssen dazu so interpretiert werden, als stünden sie hinter einem Komma!</u>

Wie berechnet man nun von einer Dezimalzahl die binäre GPZ-Näherung?

- Dezimalzahl durch 2^x, so dass 2^x die größte Zweierpotenz ist, die unter der Dezimalzahl liegt, oder ihr im Idealfall entspricht. z.B. 12 => 2^3. ==> Exponent = 3!

Mantisse = (Dezimalzahl / 2^x)-1

- Jetzt nur noch beide Zahlen ins Binärformat umwandeln.

Der Exponent wird bei IEEE in der sog. Bias-Form gespeichert. d.h. Man rechnet den Wert des Exponenten aus, als wäre er eine ganz gewöhnliche Binärzahl. Anschließend zieht man den Bias ab, fertig! Umgekehrt, will man aus einer Dezimalzahl die Bias-Darstellung generieren, addiert man zu ihr den BIAS und wandelt das Ergebniss in eine normale Binärzahl (nicht 2k!) um.

Wie wandelt man die Mantisse ins Binärformat? Nehmen wir als anschauliches Beispiel eine Mantisse von 0,75 an.

Daraus folgt: $2^{-1} = 0{,}5$ (½)
$2^{-2} = 0{,}25$ (¼)

$2^{-1} + 2^{-2} = 0{,}75$

Die Mantisse ist also binär 110000.... (entsprechend Nuller nachfüllen, wenn nötig)

Hier hatten wir scheinbar keine feste Regel, darum ist ausprobieren angesagt!!!

Problem: Wenn wir uns immer die 1 vorm Komma dazu denken, ohne sie zu speichern, wie stellen wir dann Zahlen mit einer Null vorm Komma da? Ganz einfach: Wenn der Exponent die kleinste Zahl enthält, die man mit ihm darstellen kann (egal ob 2k oder BIAS, hauptsache, die kleinste Zahl!), dann wird die Mantisse als nicht normiert betrachtet. D.h. Statt der 1 haben wir eine 0 vorm Komma. Nur so lässt sich die NULL als GPZ darstellen, indem alle Bits der Mantisse auf 0 gesetzt werden, und der Exponent die kleinst mögliche Zahl enthält.

HTH, Dennis