查表法计算 $log_2(x+1)$

杨坤泽 2023210799

October 2023

1 计算方法

根据对数函数的特点,先直接分析 $log_2(x+1)$,有 $max[log_2(x+1)-x] = 0.0861 < 2^{-3}$, 即直接对该对数函数进行查表可将幅值压缩 3bits。因此，为了利用该对数函数形式的特点，考虑将该函数拆解为多个 $log_2(1+f(x))$ 的子式进行幅值压缩后求和的查表计算。

采用与 sin(x) 类似的方法，令 $x = A + B$，其中 A 为 P 的高 6 位表示的小数，B 为 P 的低 6 位表示的小数。对于对数函数，拆解为求和项需要对内部函数式拆分为乘项，则有:

$$\begin{aligned}log_2(x+1) &= log_2(A+B+1)\\ &= log_2((\frac{A+B+1}{1+A})(1+A))\\ &= log_2(1+\frac{B}{1+A}) + log_2(1+A)\end{aligned}$$

其中 $0 \approx 2^{-13} < \frac{B}{1+A} < 1$，并没有实现缩小 ROM 存储空间的目标。

因此进一步令 $x = A + B + C$，其中 A 为 P 的高 4 位表示的小数，B 为 P 的中间 4 位表示的小数，C 为 P 的低 4 位表示的小数。则有:

$$\begin{aligned}log_2(x+1) &= log_2(A+B+C+1)\\ &= log_2((\frac{A+B+C+1}{1+A+B})(1+A+B))\\ &= log_2(1+\frac{C}{1+A+B}) + log_2(1+A+B)\end{aligned}$$

此时 $0 \approx 2^{-13} < \frac{C}{1+A+B} < 2^{-8}$，计算得到 $max[log_2(1+\frac{C}{1+A+B}) - \frac{C}{1+A+B}] = 0.0017 < 2^{-9}$，则通过拆解可以将对数函数中的一项的幅值缩减 9bits。但注意到需要计算 $\frac{C}{1+A+B}$，引入除法器增加系统的复杂性；因此考虑到 A 与 B 的值较小，直接将原式估计为 $log_2(1+C)$，采用幅值缩减的方法，有 $max[log_2(1+C) - C] = 0.0016 < 2^{-9}$，可缩减 9bits。即有

$$log_2(x+1) = log_2(1+C) + log_2(1+A+B)$$

而对于 $log_2(1+A+B)$，进行幅值压缩，计算得到 $max[log_2(1+A+B) - (A+B)] = 0.0861 < 2^{-3}$，可以缩减 3bits。计算 ROM 容量，对于前者，

地址长度为 4bits，数据长度 3bits；对于后者，地址长度 8bits，数据长度 9bits，则总的 ROM 容量为

$$ROM = 2^4 \times 3 + 2^8 \times 9 = 2352bits$$

2 仿真结果

利用 MATLAB 仿真得到计算的精度结果与曲线对比图如下：

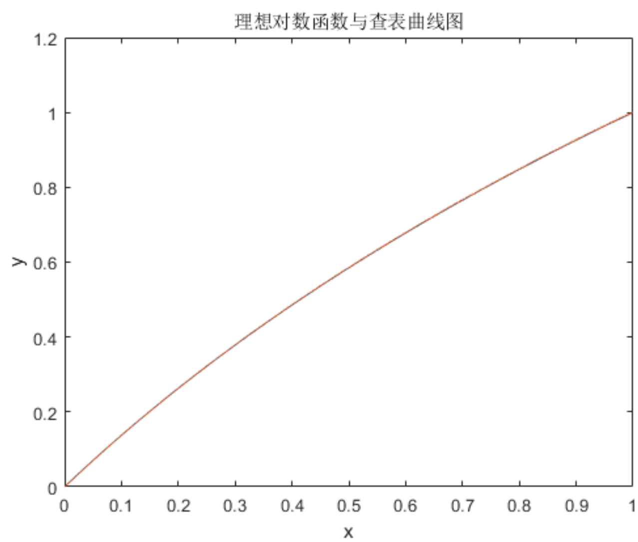

图 1: 理想对数函数与查表曲线图

注意到误差曲线稍稍超过-10，即精度不满足题目要求。原因在于 $log_2(1+\frac{C}{1+A+B})$ 的估计值误差较大，影响到了精度。因此重新计算不对其估计的结果，采用直接查表法规避引入除法器；同时注意到其存储位**宽为5，则对应地址仅需要5位即可，**ROM 容量扩大为

$$ROM = 2^8 \times 9 + 2^5 \times 5 = 2464bits$$

此时得到的对数误差结果与曲线对比图如下：

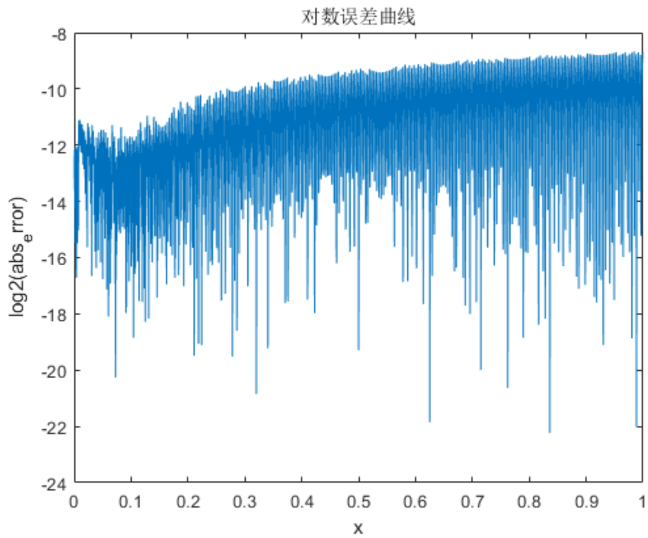

图 2: 对数误差曲线图

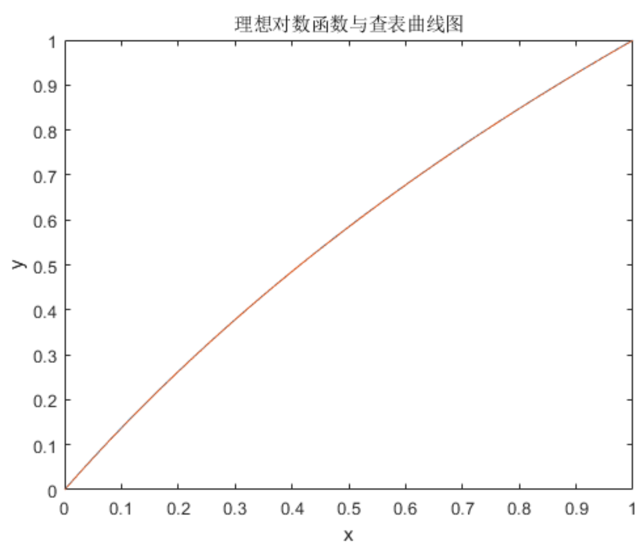

图 3: 理想对数函数与查表曲线图

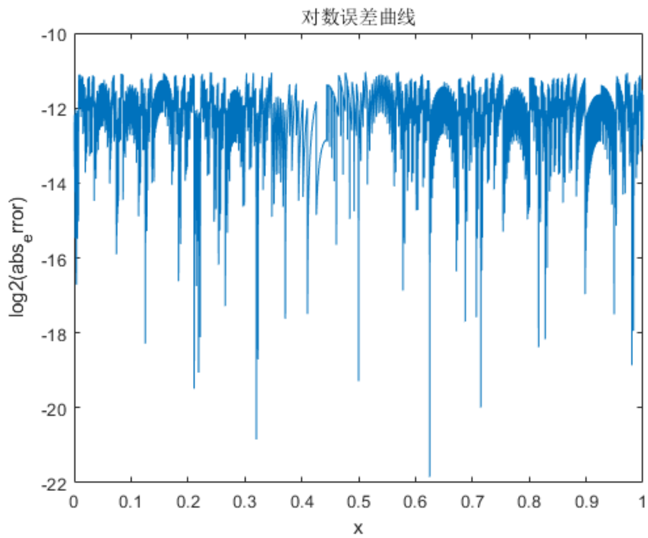

图 4: 对数误差曲线图

可以注意到误差曲线小于-10，说明满足要求；但同时付出了 ROM 容量扩大的代价，说明在实际系统设计中存在容量与精度的矛盾关系。需要根据实际应用中的精度要求灵活设计，调整 ROM 的大小，以达到相对最优解。设计的电路图结构如下：

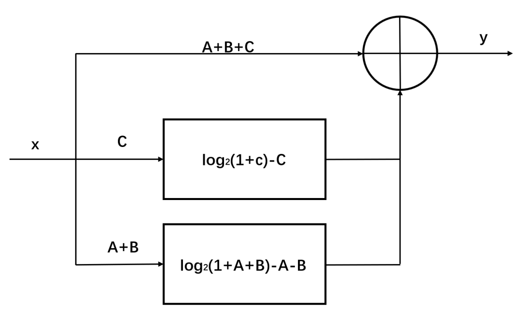

图 5: 电路结构图