Parabola Volume 32, Issue 3 (1996)

DIVISION

Geoffrey Coombs[1]

1 Introduction

It was recently found that some versions of the Intel Pentium processor possessed a flaw in the floating point divide unit. This generated considerable interest in the algorithm used in this chip to perform the division of two real numbers. In this article I intend to describe this algorithm and show that it is based on the simple long-division algorithm that should be familiar to everyone. The novel aspects of the algorithm are the techniques used to simplify the arithmetic operations involved in performing the individual steps. The origin of the flaw will be described.

The operation of division is usually written in the form

$$q = \frac{p}{d},$$

where p, the number to be divided, is referred to as the *dividend*, d is the *divisor* and the result of the operation, q, is the *quotient*.

For example, suppose we wish to divide the number 9.17 by 2.84. The standard procedure for doing this by hand is usually referred to as *long division*. The calculation is conventionally presented in the following tabular form:

$$\begin{array}{rr} & \underline{3.2} \\ 2.84 &)\ 9.17 \\ & \underline{8.52} \\ & 0.65 \end{array}$$

The most significant digit of the quotient is determined first. It is determined by a process of trial and error as being the largest multiple of the divisor that does not exceed the dividend. In this case it is 3. To determine the next quotient digit we calculate the *partial remainder* which is the difference between the dividend and the product of the previous quotient digit and the divisor. So we have

$$9.17 - 3 \times 2.84 = 9.17 - 8.52 = 0.65.$$

We then repeat the process used to determine the first quotient digit with the dividend replaced by the *shifted partial remainder*, $10 \times 0.65 = 6.5$. The quotient digit is found to be 2 and the new partial remainder is

$$6.5 - 2 \times 2.284 = 0.82.$$

[1]Geoffrey Coombs is a member of the School of Mathematics at the University of N.S.W.

It is more convenient to record the shifted partial remainders in this tabular presentation of long division process. Thus the computations required to obtain five digits of the quotient may be presented as follows:

$$
\begin{array}{rl}
 & 3.2288 \\ \cline{2-2}
2.84 \quad) & 9.17 \\
 & 8.52 \\ \cline{2-2}
 & 6.5 \\
 & 5.68 \\ \cline{2-2}
 & 8.2 \\
 & 5.68 \\ \cline{2-2}
 & 25.2 \\
 & 22.72 \\ \cline{2-2}
 & 24.8 \\
 & 22.72 \\ \cline{2-2}
 & 20.8
\end{array}
$$

In what follows we shall assume that both divisor and dividend are positive. For zero divisor the operation is undefined and for zero dividend the result is trivially zero. For negative divisor or dividend the sign of the quotient may be determined separately and the quantities may be regarded as positive for the purposes of the actual division process. Also we shall assume that the divisor and dividend lie in the range $1 \leq n < 10$. This is a minor restriction since any real number may be represented in *normalised* form. That is the most significant digit of the number is placed immediately to the left of the decimal point and a compensating power-of-ten factor is included. For example

$$
\begin{aligned}
126.56 &= 1.2656 \times 10^2, \\
0.000975 &= 9.75 \times 10^{-4}.
\end{aligned}
$$

The appropriate power of ten to include in the quotient may be obtained by simple subtraction.

The part of a normalised number that contains the actual digits is referred to as the *significand*.

2 The Algebraic Representation of the Division Process.

If the divisor and dividend satisfy the requirements specified above, the quotient may be represented by

$$q = q_0.q_1 q_2 q_3 \ldots,$$

where the quotient digits q_j satisfy $0 \leq q_j < 10$. Now the expression for the computation of the j^{th} partial remainder, p_j, is

$$p_{j+1} = 10 \times p_j - q_j d.$$

So the procedure for performing the division process may be expressed as:

- Given the partial remainder p_j

– Form the shifted partial remainder $10 \times p_j$,

– Determine the quotient digit q_j.

– Calculate the product $q_j d$,

– Perform the subtraction $10 \times p_j - q_j d \ \ (= p_{j+1})$,

- Repeat these steps until the required number of quotient digits are obtained.

So that this schema is consistent with the example discussed above we must take the initial partial remainder p_0 as the dividend divided by 10, that is $p_0 = p/10$.

Of the steps involved in this process, the shifting of the partial remainder is trivial and the subtraction is straightforward. This leaves the formation of the product $q_{j-1}d$ and the determination of the quotient digit q_j as computationally difficult tasks. We consider first the formation of the product.

3 The representation of numbers and simplified multiplication.

As we have seen the significand of a number, q say, is represented in terms of its constituent digits q_j by

$$q = q_0 . q_1 q_2 q_3 \ldots ,$$

where $0 \leq q_j < 10$ for $j > 0$ and $1 \leq q_0 < 10$. The actual value of the number is

$$\begin{aligned} q &= q_0 + q_1 \times 10^{-1} + q_2 \times 10^{-2} + q_3 \times 10^{-3} + \cdots , \\ &= \sum_{j=0}^{\infty} q_j \times 10^{-j}. \end{aligned}$$

Suppose we extend the range of values of the allowed digits to include negative values. That is we require that $-10 < q_i < 10$. The presence of a negative digit in the usual representation of a number is indicated by a bar, so the number $2.3\bar{4}6$ has the value

$$2.3\bar{4}6 = 2 + 3 \times 10^{-1} + (-4) \times 10^{-2} + 6 \times 10^{-3}.$$

This can always be converted into the usual form by the following manipulations:

$$\begin{aligned} 2.3\bar{4}6 &= 2 + 3 \times 10^{-1} + (-4) \times 10^{-2} + 6 \times 10^{-3}, \\ &= 2 + 3 \times 10^{-1} + (6 - 10) \times 10^{-2} + 6 \times 10^{-3}, \\ &= 2 + 3 \times 10^{-1} - 10 \times 10^{-2} + 6 \times 10^{-2} + 6 \times 10^{-3}, \\ &= 2 + (3 - 1) \times 10^{-1} + 6 \times 10^{-2} + 6 \times 10^{-3}, \\ &= 2 + 2 \times 10^{-1} + 6 \times 10^{-2} + 6 \times 10^{-3}, \\ &= 2.266. \end{aligned}$$

This illustrates a property of our extended representation: there is more than one representation of each number. We can use this to bring about a simplification. It is clear that we can dispense with the digits $6, 7, 8, 9$. As for any digit, q_j say, greater than 5 may be replaced by $10 - q_j$ which is less than or equal to five. For example

$$2.736 = 2.72\bar{4} = 1.\bar{3}2\bar{4}.$$

or

$$5.892 = 5.7\bar{1}2 = 4.\bar{3}\bar{1}2.$$

The advantage of this lies in the fact that we need only know how to multiply numbers up to 5 rather than up to 9. (Multiplication by 10 being trivial.) We may further reduce the required knowledge of multiplication tables by representing the numbers with respect to a smaller radix (or base).

For general radix r, the value of the number n having the representation

$$n = n_0.n_1n_2n_3 \cdots ,$$

the value is

$$n = \sum_{j=0}^{\infty} n_j \times r^{-j},$$

where r is usually a positive integer and $0 \leq n_j < r$ for $j > 0$ and $1 \leq n_0 < r$. This is a generalisation of the radix 10 representation introduced above.

By choosing as our radix $r = 4$ we only need to know how to multiply 0,1 and 2. The drawback is that we achieve a smaller improvement in precision for each iteration of the method. However arithmetic involving numbers no larger than 2 lends itself to digital machine computation and thus the radix-4 method is to be preferred.

We turn now to the problem of the determination of the quotient digits.

4 Derivation of an Important Inequality.

For general radix r the relation between the quotient and the quotient digits is

$$q = \frac{p}{d} = \sum_{k=0}^{\infty} r^{-k} q_k.$$

Consideration of this relation and the expression for the partial remainder p_j suggests that we can write

$$p_j = \sum_{k=j}^{\infty} r^{-k+j-1} q_k d.$$

This form certainly has the properties so far established for p_j:

$$\begin{aligned} p_0 &= \sum_{k=0}^{\infty} r^{-k-1} q_k d, \\ &= \frac{d}{r} \sum_{k=0}^{\infty} r^{-k} q_k, \\ &= \frac{d}{r} \frac{p}{d}, \\ &= \frac{p}{r}. \end{aligned}$$

And

$$\begin{aligned} p_j &= \sum_{k=j}^{\infty} r^{-k+j-1} q_k d, \\ &= r^{-1} d q_j + \sum_{k=j+1}^{\infty} r^{-k+j-1} q_k d, \\ &= r^{-1} d q_j + r^{-1} \sum_{k=j+1}^{\infty} r^{-k+(j+1)-1} q_k d, \\ &= r^{-1} d q_j + r^{-1} p_{j+1}, \end{aligned}$$

and so

$$p_{j+1} = r p_j - d q_j.$$

Although the closed form for the partial remainder obtained here is useless for computational purposes, it does allow us to derive an extremely useful theoretical result. Suppose the maximum value of the quotient digits is q_{max}: that is $q_i \leq q_{max}$ for all i. We can thus obtain an upper bound on the partial remainder,

$$
\begin{aligned}
p_j &= \sum_{k=j}^{\infty} r^{-k+j-1} q_k d, \\
&\leq \sum_{k=j}^{\infty} r^{-k+j-1} q_{max} d, \\
&= q_{max} d \sum_{k=j}^{\infty} r^{-k+j-1}, \\
&= q_{max} d \sum_{i=0}^{\infty} r^{-i-1}, \\
&= q_{max} d r^{-1} \sum_{i=0}^{\infty} r^{-i}, \\
&= q_{max} d r^{-1} \frac{1}{1-r^{-1}}, \\
&= \frac{q_{max} d}{r-1}.
\end{aligned}
$$

So

$$p_j \leq \frac{q_{max} d}{r-1},$$

and similarly, if q_{min} is the minimum possible value of the quotient digits, then by a similar argument it follows that

$$p_j \geq \frac{q_{min} d}{r-1},$$

or

$$\frac{q_{min} d}{r-1} \leq p_j \leq \frac{q_{max} d}{r-1}.$$

For case of the number system with radix $r = 4$ and using a representation in which negative digits are included and are drawn from the set $0, \pm 1, \pm 2$, this inequality becomes

$$-\frac{2}{3} d \leq p_j \leq \frac{2}{3} d.$$

Rearranging the standard relation to give

$$r p_j = p_{j+1} + q_j d,$$

and noting that the above inequality is also true for p_{j+1}, we obtain the further inequality

$$\left(-\frac{2}{3} + q_j\right) d \leq r p_j \leq \left(\frac{2}{3} + q_j\right) d.$$

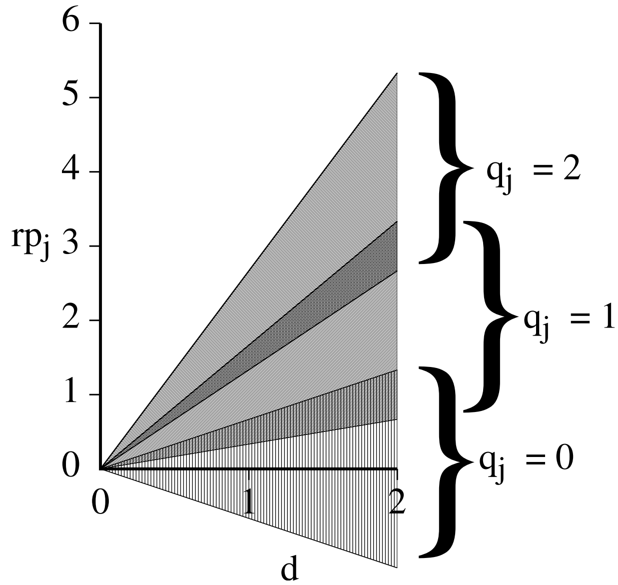

Figure 1: The P-D plot.

5 Determination of the Quotient Digits.

We may regard this inequality as representing a region of the d–rp_j plane for a fixed value of the quotient digit q_j. For example, for $q_j = 1$ the inequality becomes

$$\frac{1}{3}d \leq rp_j \leq \frac{5}{3}d,$$

and the region is that part of the plane that lies between the straight lines $rp_j = \frac{1}{3}d$ and $rp_j = \frac{5}{3}d$.

There will be a region of the d–rp_j plane for each of the five possible values of the quotient digit: $0, \pm 1, \pm 2$. A portion of this plane is shown in the P-D plot of Figure 1. Note that these regions overlap in part.

For the sake of simplicity the parts of this plot dealing with negative divisors and negative quotient digits have been omitted. It should be obvious that the P-D plot is symmetric about both axes.

To determine the quotient digit in the current iteration, the point in the P-D plane that corresponds to the divisor and the current value of the shifted partial remainder is located and the quotient digit for that region may be read off. Due to overlap of these regions there may be a choice of digits but rather than being a complication, this phenomenon may be exploited in the machine implementation of this process.

It would appear that to determine the appropriate region some form of calculation is required. However we can reduce this to a table look-up by relaxing the precision with which the divisor and the shifted partial remainder are specified.

First we note that the value of a positive divisor must be no less than one, as we have assumed it is normalised, and less than two, as we are treating it as an effectively

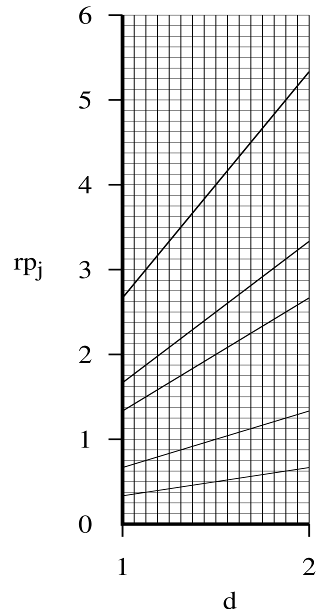

Figure 2: The P-D plot showing the discretization.

radix 2 quantity. We subdivide this interval into 16 boxes of equal width and label them with the numbers 1 through $1\frac{15}{16}$. This corresponds to the radix 2 numbers 1.0000 through 1.1111. The vertical, or shifted partial remainder, axis is divided up into boxes of width 1/8 or 0.001.

Thus the d–rp_j plane is divided up into a set of rectangles, each of which, apart from those rectangles that straddle two regions, has a unique value of the quotient digit associated with it. For the rectangles that straddle regions a decision is made based on some criterion as to which of the two possible values they belong. This discretization of the P-D plot is shown if Figure 2.

Thus we have a table that may be consulted to determine the quotient digit on the basis of the sampling of the first few bits of the dividend and the shifted partial remainder.

6 Floating-Point Division in the Pentium Processor

It is just this algorithm for division that is used in the Intel Pentium microprocessor. The look-up table is implemented directly on the chip. It was found that some versions of the chip were flawed in that some divide operations yielded incorrect results. The fault was traced to five incorrect entries in the look-up table.

The following may be of interest:[2]

Intel's web site:
`http://pentium.intel.com/procs/support/pentium/fdiv/index.htm`

An article from Byte magazine:
`http://byte.com/art/9503/sec13/art1.htm`

A collection of documents relating to the discovery of the Pentium bug:
`http://www.mathworks.com/README.html`

* * * * * * * * *

A CARD PREDICTION

You ask someone else to shuffle a pack of cards and then write down the name of a card on a piece of paper which is placed aside.

Deal twelve cards face down, ask some-one to turn four of the cards over and place the rest of the dealt cards on the **bottom** of the pack. We will suppose that the four cards are a $3,\ 7,\ 10$ and King. Ask some-one to assign a value to the King (which we will suppose to be 5). Now deal out $10-3=7$ cards alongside the $3,\ 10-7=3$ cards alongside the 7, no cards $(10-10=0)$ alongside the 10 and $10-5=5$ cards alongside the King.

Now add the value of the cards chosen (in this example, $3+7+10+5=25$) and ask some-one to find the corresponding card (here the 25th) in the pack. It will be the card which you wrote on the piece of paper.

Method in solutions section

[2]Editorial note, February 2014: these links are now dead