HIGHER RADIX ON-LINE DIVISION*

Kishor S. Trivedi
and
Joseph G. Rusnak

Department of Computer Science
Duke University
Durham, N.C., 27706

Abstract

We present a formal proof of correctness of the on-line division algorithm specified in an earlier paper [1]. We also derive two radix 4 on-line division algorithms, with non-redundant and redundant operands respectively.

1. Introduction

By on-line arithmetic algorithms we mean those arithmetic algorithms in which the operands as well as the result flow through the arithmetic unit in a digit-by-digit fashion, in order of significance [1]. It is assumed that δ initial digits of the operands are needed to produce the first digit of the result. Thereafter, one digit of the result is produced upon receiving one digit of each of the operands. Here δ is a small positive integer, called the on-line delay.

Consider an m-digit radix r number $N = \sum_{i=1}^{m} n_i r^{-i}$.

In the conventional representation, each digit n_i can take on any value from the digit set $\{0,1,\ldots,r-1\}$. Such representations which allow only r values in the digit set, are non-redundant since there is a unique representation for each (representable) number. By contrast, number systems that allow more than r values in the digit set are redundant. Redundant number representations are often useful in speeding up arithmetic operations [2], [3]. It is not difficult to see that the use of redundant number representation is mandatory for on-line algorithms. If we were to use a non-redundant number system, then even for simple operations like addition and subtraction there is an on-line delay of δ=m due to carry propagation. If we allow redundancy in the number representation, then it is possible to limit carry propagation to one digit position. On-line algorithms for addition, substraction, and multiplication have been developed with δ=1 [1], [3], [4]. In this paper, we develop two radix 4 on-line division algorithms with δ=3 and δ=5 respectively. In general, the value of δ for an on-line division algorithm depends upon the radix and the properties of the digit set employed.

The concept of on-line arithmetic is potentially useful in variable precision and multi-precision arithmetic, in significance arithmetic [5], and in digitally chained [6] and pipelined [7] arithmetic units. It also facilitates the modular design of an arithmetic unit. Detailed study of arithmetic unit architectures which exploit the features of on-line arithmetic algorithms is a topic of future research, although some initial work has been done [8]. Ercegovac [9] has developed a general on-line method for computing a large class of functions. Ercegovac [10] has recently developed an on-line square-rooting algorithm.

In reference [1], an on-line algorithm for binary division was specified. A procedure for deriving a higher radix on-line division algorithm for redundant operands was also specified. Irwin [8] has studied higher radix on-line division with non-redundant operands. She allowed the partial remainder to be in a redundant format so that carry-free addition could be utilized in each interative step.

The reason for considering higher radix arithmetic operations is a possible gain in efficiency, since the use of a higher radix causes a corresponding reduction in the number of iterations required (although at the time the complexity of each individual iteration is increased). For a discussion of conventional higher radix division algorithms see [2,11,12,13,14].

The purpose of this paper is twofold. First, in sections 1 and 2, we describe and give a formal proof of correctness of on-line division. Secondly, in sections 3 and 4 we derive two radix 4 on-line division algorithms, with non-redundant and redundant operands respectively.

2. The On-Line Division Schema

The following schema for on-line division was first given in [1]. Note that this schema defines a class of division algorithms. This schema produces an on-line division algorithm only when given an interpretation by specifying the representations of N, D, P, and Q, and by specifying the selection function SELECT.

*Supported in part under the National Science Foundation Grant No. US NSF 76-24417.

SCHEMA OD [On-Line Division]

Step 1 [Initialization]: $\hat{P}_0 \leftarrow \sum_{i=1}^{\delta} n_i r^{-i}$; $\hat{D}_0 \leftarrow \sum_{i=1}^{\delta} d_i r^{-i}$; $j \leftarrow 0$;

Step 2 [Loop]: while $(j<m)$ do

begin

Step 2.1 [Selection]: $q_{j+1} \leftarrow \text{SELECT}\ (r\hat{P}_j, \hat{D}_j)$;

Step 2.2 [Recursions]: $\hat{D}_{j+1} \leftarrow \hat{D}_j + d_{j+\delta+1} r^{-j-\delta-1}$;

$$\hat{P}_{j+1} \leftarrow r\hat{P}_j - q_{j+1}\hat{D}_{j+1} + n_{j+\delta+1} r^{-\delta} - \left(\sum_{i=1}^{j} q_i r^{-i}\right) d_{j+\delta+1} r^{-\delta};$$

Step 2.3 [Increment]: $j \leftarrow j+1$

end.

The value of r is the radix, and δ represents the assumed on-line delay of algorithm. The function SELECT in step 2.1 represents the quotient digit selection procedure. In step 1, we accumulate the δ leading digits of the dividend in $\hat{P}_0$ and δ leading digits of the divisor in $\hat{D}_0$. After appropriately selecting the quotient digit q_{j+1} in step 2.1, we proceed to insert (in step 2.2) the newly arrived digit of the divisor into the current approximation $\hat{D}_{j+1}$ to the divisor. The new digit of the dividend $n_{j+\delta+1}$ is simply placed in the appropriate place (i.e., with the weight $r^{-j-\delta-1}$). The formation of the next partial remainder $\hat{P}_{j+1}$ is more complex than in conventional division, since we have to account for the newly-arriving digits of the operands. The last term on the right hand side of the formula for $\hat{P}_{j+1}$ accounts for the interaction of the new digit of the divisor with the previously emitted digits of the quotient. The condition test $(j<m)$ of the while loop assures us that we will generate the quotient to m-digit precision. We note that it is possible to extend the above schema to handle the case where the dividend and divisor are available to m_1 -digit and m_2 -digit precision, respectively, and the quotient is generated to m_3 -digit precision.

3. Correctness Of The Schema

Now we proceed to prove the correctness of the schema OD. We will follow the notation as in [15]. First consider the termination aspect of correctness. Note that the while loop in the schema is executed a fixed (m) number of times. Thus if we can show that the function SELECT always terminates (as we will show in a subsequent section), then we can conclude that schema OD will always terminate.

Next consider the second aspect of the correctness of schema OD: whenever it terminates, it generates the quotient $Q = \sum_{i=1}^{m} q_i r^{-i}$ which is an m-digit approximation to N/D, where the divided is $N = \sum_{i=1}^{m} n_i r^{-i}$ and the divisor is $D = \sum_{i=1}^{m} d_i r^{-i}$.

Formally, the desired output assertion is

$$A_1 : \{Q = N/D - \alpha\, r^{-m},\ 1/2 \le \alpha \le 1\}$$

We desire to show that

true {SCHEMA OD} A_1 .

To achieve this desired result, let us define the loop assertion

$$A_2: \left\{\hat{P}_j = r^j \left[\sum_{i=1}^{j+\delta} n_i r^{-i} - \left(\sum_{i=1}^{j} q_i r^{-i}\right)\left(\sum_{i=1}^{j+\delta} d_i r^{-i}\right)\right]\right\} \quad (3.1)$$

Lemma 1: true {step 1} $A_2 \wedge (j=0)$

Proof: By substituting j=0 in the assertion A_2, we get, $\hat{P}_0 = \sum_{i=1}^{\delta} n_i r^{-i}$. But this coincides with the first assignment in Step 1.

Lemma 2: $A_2 \wedge (j < m)$ {Step 2.1; Step 2.2; Step 2.3} A_2

Proof: (follows by direct substitution of the recursions of Step 2.2 and simplification).

Now using lemma 2 and the rule of iteration from Hoare's paper [15], we can prove the following

Lemma 3: A_2 {Step 2} $\neg(j < m) \wedge A_2$.

From the rule of composition from Hoare's paper [15], and Lemma 1 and 3, we can prove

Lemma 4: true {SCHEMA OD} $\neg(j < m) \wedge A_2$.

In fact, upon termination of the schema, we have $(j = m) \wedge A_2$, or

$$\hat{P}_m = r^m \left[\sum_{i=1}^{m+\delta} n_i r^{-i} - \left(\sum_{i=1}^{m} q_i r^{-i} \right) \left(\sum_{i=1}^{m+\delta} d_i r^{-i} \right) \right]$$

If we assume that N and D are padded on the right with δ zeros, then the above gives us:

$$\hat{P}_m = r^m [N - QD]$$

or

$$Q = N/D - r^{-m}(\hat{P}_m/D) \tag{3.2}$$

This will imply the output assertion A_1 provided that $(\hat{P}_m/D)$ can be bounded.

We therefore define the additional loop assertion A_3 to be:

$$A_3: \{|\hat{P}_j \leq \alpha D\}$$

Following Robertson [2], we require that $1/2 \leq \alpha \leq 1$. Note that the value of α is determined by the properties of the number system employed [2]. It is clear from the above discussion that the following lemma holds:

Lemma 5: $A_2 \wedge A_3 \wedge (j=m) \supset A_1$.

To complete the proof of correctness of schema OD, we need to prove the following two lemmas:

Lemma 6: true {Step 1} $A_3 \wedge (j = 0)$.

Lemma 7: $A_3 \wedge (j < m)$ {Step 2.1; Step 2.2; Step 2.3} A_3

Lemma 6 can be satisfied if we assume that the range restriction implied by assertion A_3 is initially enforced by appropriately preshifting the dividend. Lemma 7 gives the input-output specifications for the selection function SELECT and will be satisfied in the course of deriving the selection function for our two examples of radix 4 on-line division. We can now prove the following theorem.

Theorem 1: Any interpretation of the schema OD results in a correct on-line division algorithm provided that a selection function is defined which satisfies the conditions of Lemma 7 and provided that the initial preshifting of the dividend is carried out so as to satisfy the condition of Lemma 6.

Proof: Using lemma 7 and the rule of iteration [15], we conclude that

$$A_3 \ \{\text{Step 2}\} \neg (j < m) \wedge A_3.$$

Then from the above and lemma 6, we have

$$\underline{\text{true}} \ \{\text{SCHEMA OD}\} \neg (j < m) \wedge A_3,$$

by the rule of composition [15]. Combining this with lemma 4, we have

$$\underline{\text{true}} \ \{\text{SCHEMA OD}\} \neg (j < m) \wedge A_2 \wedge A_3.$$

In fact, upon termination of schema OD, we have

$$(j = m) \wedge A_2 \wedge A_3.$$

Then from lemma 5 and the rule of consequence [15] we conclude that

$$\underline{\text{true}} \ \{\text{SCHEMA OD}\} \ A_1$$

which was to be shown.

4. Radix 4 Division with Non-Redundant Operands

We give the division schema OD an interpretation as follows. Let $r=4$, and let $n_i, d_i \in \{0,1,2,3\}$. Let $q_i \in \{\bar{2},\bar{1},0,1,2\}$. The input assertion for the selection function is that the partial remainder $\hat{P}_j$ satisfies the range restriction specified by assertion A_3. It is necessary to derive a selection function which will produce the quotient digit $q_{j+1} \in \{\bar{2},\bar{1},0,1,2\}$ such that the resulting $\hat{P}_{j+1}$ (after the execution of Step 2.2) also satisfies the range restriction of assertion A_3. Now if we can arrange for $\hat{P}_0$ to satisfy the assertion A_3 (by appropriately preshifting the dividend), then by induction, the assertion A_3 will be satisfied for all values of j, $0 \leq j \leq m$. This will then imply that $|\hat{P}_m/D| \leq \alpha$ which in turn implies that the algorithm derived from schema OD produces the quotient Q to m-digit precision.

Our approach to deriving a selection function satisfying the above properties parallels the standard reverse approach [2,14]. Assuming $\hat{P}_{j+1}$ must satisfy the assertion A_3, we obtain the corresponding range of $\hat{P}_j$ for each possible value of q_{j+1}. This range of value of $\hat{P}_j$ is called the q_{j+1} selection region, and is conveniently represented on a so-called P-D plot[14]. The actual selection function is then derived from the P-D plot.

Unlike the case of a conventional division algorithm, deriving the q_{j+1} selection regions for on-line division is quite difficult if approached directly due to the complexity of the recursion for $\hat{P}_{j+1}$. We follow an interesting indirect approach to this problem as in [1]. We assume that a simulation of a conventional division algorithm (derived from the following division schema CD) is being performed synchronously with the execution of the on-line division algorithm. This simulation does not actually occur; its only purpose is to ease the derivation of the selection function. Indeed, this simulation could not be performed in an on-line environment even if desired, since it requires that all of the digits of the two operands be available in advance.

We postulate that after a quotient digit is selected in Step 2.1, we execute Step 2.2 of the simulated conventional division algorithm. Assuming that the initial value of P_j (in the simulated algorithm) satisfies assertion A_3, we require that the resulting P_{j+1} also satisfy assertion A_3. We note that determining appropriate q_{j+1} selection regions for P_j is relatively easy [2]. Since P_j is not known, we derive a relation between P_j and $\hat{P}_j$ and then transform each q_{j+1} selection region of P_j into a q_{j+1} selection region for $\hat{P}_j$.

Therefore, we consider the conventional division schema, some of whose interpretations yield SRT division, restoring division, and non-restoring division.

<u>SCHEMA CD[Conventional Division]</u>

Step 1 [Initialization]: $P_0 \leftarrow N;\ j \leftarrow 0;$

Step 2 [Loop]: <u>While</u> $(j<m)$ <u>do</u>

<u>begin</u>

Step 2.1 [Selection]: $q_{j+1} \leftarrow \text{SELECT}\ (rP_j, D);$

Step 2.2 [Recursion]: $P_{j+1} \leftarrow rP_j - q_{j+1}D;$

Step 2.3 [Increment]: $j \leftarrow j+1$

<u>end</u>.

Using induction on j, we derive the following loop invariant for the conventional division schema.

$$P_j = r^j \left[\sum_{i=1}^{m} n_i r^{-i} - \left(\sum_{i=1}^{j} q_i r^{-i} \right) \left(\sum_{i=1}^{m} d_i r^{-i} \right) \right] \tag{4.1}$$

From (3.1) and (4.1) we get

$$P_j - \hat{P}_j = r^j \left[\sum_{i=j+\delta+1}^{m} n_i r^{-i} - \left(\sum_{i=1}^{j} q_i r^{-i} \right) \left(\sum_{i=j+\delta+1}^{m} d_i r^{-i} \right) \right] \tag{4.2}$$

Since $n_i, d_i \in \{0,1,2,3\}$ and $q_i \in \{\bar{2},\bar{1},0,1,2\}$, we see that

$$P_j - \hat{P}_j \le r^j \left[\sum_{i=j+\delta+1}^{m} 3r^{-i} + \left(\sum_{i=1}^{j} 2r^{-i} \right) \left(\sum_{i=j+\delta+1}^{m} 3r^{-i} \right) \right] \tag{4.3}$$

and

$$P_j - \hat{P}_j \le -r^j \left[\left(\sum_{i=1}^{j} 2r^{-i} \right) \left(\sum_{i=j+\delta+1}^{m} 3r^{-i} \right) \right] \tag{4.4}$$

Letting $m \to \infty$ in (4.3) we set

$$P_j - \hat{P}_j \le r^j \left[\frac{3r^{-j-\delta-1}}{1-r^{-1}} \left(\frac{2r^{-1}(1-r^{j})}{1-r^{-1}} \right) \left(\frac{3r^{-j-\delta-1}}{1-r^{-1}} \right) \right]$$

$$\le r^{-\delta} + \left(\frac{2r}{3} \right) \left(1-r^{-j} \right) r^{-\delta-1}$$

Now letting $j \to \infty$,

$$P_j - \hat{P}_j \le r^{-\delta} + \frac{2}{3} r^{-\delta}$$

$$\le \left(1 + 2/3 \right) r^{-\delta} \tag{4.5}$$

Subjecting (4.4) to similar manipulation, we obtain

$$P_j - \hat{P}_j \ge -\frac{2}{3} r^{-\delta} \tag{4.6}$$

Therefore,

$$-\frac{2}{3} r^{-\delta} \le P_j - \hat{P}_j \le \frac{5}{3} r^{-\delta} \tag{4.7}$$

Now from Robertson's paper [2] we know that the range restriction on P_j is:

$$-\frac{2}{3} D \le P_j \le \frac{2}{3} D \tag{4.8}$$

Therefore, the range restriction on $\hat{P}_j$ is:

$$-\frac{2}{3} D + \frac{2}{3} r^{-\delta} \le \hat{P}_j \le \frac{2}{3} D - \frac{5}{3} r^{-\delta} \tag{4.9}$$

From [2,14], the selection region for a quotient digit $q_{j+1} = i$ for the conventional division schema is defined (in our example) as:

$$-\left(\frac{2}{3} + i \right) D \le rP_j \le \left(\frac{2}{3} + i \right) D \tag{4.10}$$

Therefore, using (4.7), the selection region for $r\hat{P}_j$ is given by:

$$\left(-\frac{2}{3} + i \right) D + \frac{2}{3} r^{-\delta+1} \le r\hat{P}_j \le \left(\frac{2}{3} + i \right) D - \frac{5}{3} r^{-\delta+1}$$

Assume without loss of generality that D>0, and that while the representation of D is with respect to radix 4, the normalization of D is with respect to radix 2. Clearly the minimum value of D is 1/2. The $\hat{P}$-D plot of the selection regions is shown in Figure 1. Note how the adjacent selection regions overlap. This is due to the redundancy in the representation of Q, and is the reason why we can make correct choices for quotient digits based upon limited information about D (which is arriving digit by digit).

In particular, we require overlap of selection regions over the entire permissible range of D, i.e., the interval [1/2, 1]. The overlap region for $q_{j+1} = 1$ is:

$$\left(-\frac{2}{3}+1\right) D + \frac{2}{3}4^{-\delta+1} \le r\hat{P}_j \le \frac{2}{3} D - \frac{5}{3} 4^{-\delta+1}$$

Clearly the worst case is at D = 1/2. To ensure overlap at D = 1/2, we have:

$$\left(-\frac{2}{3}+1\right) D + \frac{2}{3}4^{-\delta+1} < \frac{2}{3} D - \frac{5}{3} 4^{-\delta+1}$$

$$\frac{1}{6} + \frac{2}{3} 4^{-\delta+1} < \frac{2}{6} - \frac{5}{3} 4^{-\delta+1}$$

$$\frac{7}{3} 4^{-\delta+1} < \frac{1}{6}$$

$$4^{-\delta+1} < \frac{1}{14}$$

$$\delta-1 \ge 2$$

$$\delta \ge 3$$

A similar analysis holds for the overlap region for $q_{j+1} = 2$. The other cases are symmetric to these two cases.

The slanted lines of the selection regions mean that, in general, full precision comparisons of $r\hat{P}_j$ and D will be required during quotient digit selection. For the sake of efficiency, we desire to replace these slanted lines with horizontal lines representing limited precision comparisons. Clearly in Figure 1 a single horizontal line is impossible, so we compromise with a series of steps in each selection region, as illustrated in Figure 2. Computing these steps is straighforward, and produces the following selection function.

D	$\hat{P}_j$	q_{j+1}
	$\frac{25}{32} \le 4\hat{P}_j$	2
	$\frac{9}{32} \le 4\hat{P}_j \le \frac{25}{32}$	1
$\frac{1}{2} \le D \le \frac{19}{32}$	$-\frac{9}{32} \le 4\hat{P}_j \le \frac{9}{32}$	0
	$-\frac{25}{32} \le 4\hat{P}_j \le -\frac{9}{32}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{25}{32}$	$\bar{2}$
	$\frac{31}{32} \le 4\hat{P}_j$	2
	$\frac{11}{32} \le 4\hat{P}_j \le \frac{31}{32}$	1
$\frac{19}{32} \le D \le \frac{23}{32}$	$-\frac{11}{32} \le 4\hat{P}_j \le \frac{11}{32}$	0
	$-\frac{31}{32} \le 4\hat{P}_j \le -\frac{11}{32}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{31}{32}$	$\bar{2}$
	$\frac{37}{32} \le 4\hat{P}_j$	2
	$\frac{14}{32} \le 4\hat{P}_j \le \frac{37}{32}$	1
$\frac{23}{32} \le D \le \frac{27}{32}$	$-\frac{14}{32} \le 4\hat{P}_j \le \frac{14}{32}$	0
	$-\frac{37}{32} \le 4\hat{P}_j \le -\frac{14}{32}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{37}{32}$	$\bar{2}$
	$\frac{45}{32} \le 4\hat{P}_j$	2
	$\frac{17}{32} \le 4\hat{P}_j \le \frac{45}{32}$	1
$\frac{27}{32} \le D \le 1$	$-\frac{17}{32} \le 4\hat{P}_j \le \frac{17}{32}$	0
	$-\frac{45}{32} \le 4\hat{P}_j \le -\frac{17}{32}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{45}{32}$	$\bar{2}$

Note that while $\delta = 3$, this is three radix 4 digits, and thus we have 6 bits to work with in the selection function comparisons.

To conclude this section, an example of radix 4 on-line division with non-redundant operands is given.

N: 0 3 1 2 0 0 2 2 3 1 2 3 0 0 2 2

D: 2 2 1 2 3 3 1 0 3 2 3 2 0 2 3 1

$\hat{P}_j$	q_{j+1}
0.203125	1
0.1953125	1
0.1181640625	1
$\bar{0}$.1940917969	$\bar{1}$
$\bar{0}$.09808349609	$\bar{1}$
0.2910766602	2
$\bar{0}$.1083145142	$\bar{1}$
0.2244243622	1
0.2616007328	2
$\bar{0}$.2211868763	$\bar{1}$
$\bar{0}$.2325726748	$\bar{1}$
$\bar{0}$.2882312164	$\bar{2}$
0.1675017942	1
0.04402466374	0
0.176099855	1
0.052219779	0

5. Radix 4 Division With Redundant Operands

Let the radix r=4, and let

$$N = \sum_{i=1}^{m} n_i r^{-i} \qquad \text{(dividend)}$$

$$D = \sum_{i=1}^{m} d_i r^{-i} \qquad \text{(divisor)}$$

$$Q = \sum_{i=1}^{m} q_i r^{-i} \qquad \text{(quotient)}$$

where n_i, d_i, and $q_i \in \{\bar{2},\bar{1},0,1,2\}$. Thus $\rho=2$ and $K = (\rho/r-1) = 2/3$ (the degree of redundancy).

From Robertson's paper [2], we have the following bound on P_j (for the conventional division schema).

$$-\frac{2}{3} D \le P_j \le \frac{2}{3} D$$

we also have equation (4.2)

$$P_j-\hat{P}_j = r^j \left[\sum_{i=j+\delta+1}^{m} n_i r^{-i} - \left(\sum_{i=1}^{j} q_i r^{-i} \right) \left(\sum_{i=j+\delta+1}^{m} d_i r^{-i} \right) \right]$$

Because of the fact that the quotient and operands are redundant, it is easier in this case to derive a bound on the quantity $P_j - \hat{P}_j$. Since $|n_j|$, $|d_j|$, and $|q_i| < 2$, it can be shown that

$$\sum_{i=j+\delta+1}^{m} n_i r^{-i} \le 2 \left[\frac{r^{-j-\delta-1} - r^{-m-1}}{1-r^{-1}} \right]$$

$$\sum_{i=1}^{j} q_i r^{-i} \le 2 \left[\frac{r^{-1} - r^{-j-1}}{1-r^{-1}} \right]$$

$$\sum_{i=j+\delta+1}^{m} d_i r^{-i} \le 2 \left[\frac{r^{-j-\delta-1} - r^{-m-1}}{1-r^{-1}} \right]$$

Hence

$$|P_j-\hat{P}_j| \le 2r^j \left[\frac{r^{-j-\delta-1} - r^{-m-1}}{1-r^{-1}} - \left(\frac{r^{-1} - r^{-j-1}}{1-r^{-1}} \right) \left(\frac{r^{-j-\delta-1} - r^{-m-1}}{1-r^{-1}} \right) \right]$$

which simplies to

$$|P_j-\hat{P}_j| \le \left(\frac{2}{3} + \frac{4}{9} \right) 4^{-\delta}$$

after substituting the value 4 for r.

Using (5.1) and (5.2), we derive the following range restrictions on $\hat{P}_j$.

$$-\frac{2}{3} D + \left(\frac{2}{3} + \frac{4}{9} \right) 4^{-\delta} \le \hat{P}_j \le \frac{2}{3} D - \left(\frac{2}{3} + \frac{4}{9} \right) 4^{-\delta}$$

Using (4.8) and (5.2), we derive the range for the selection region of quotient digit $q_{j+1} = i$ in terms of $\hat{P}_j$.

$$\left(-\frac{2}{3} + i \right) D + \left(\frac{2}{3} + \frac{4}{9} \right) 4^{-\delta+1} \le 4\hat{P}_j \le \left(\frac{2}{3} + i \right) D - \left(\frac{2}{3} + \frac{4}{9} \right) 4^{-\delta+1}$$

Assume without loss of generality that D>0. Also assume that D is standardized as described in Atkin's thesis [14]. This means that the first digit of D is greater that 0. Clearly in our example the minimum value of D is the following:

$$D_{min} = \frac{1}{r} + \sum_{i=2}^{m} (-2) r^{-i}$$

$$\ge \frac{1}{4} - 2 \sum_{i=1}^{\infty} 4^{-i}$$

$$\ge \frac{1}{12}$$

The $\hat{P}$-D plot of the selection regions is shown in Figure 3. The fact that the quotient digits q_i are redundant provides the overlap of adjacent selection regions, and we will exploit this fact to derive a selection function which allows us to make correct quotient digit selections based on limited information about D.

As in the previous section, we wish to compute steps inside each overlap region to serve as the basis for limited-precision comparisons in our selection function. But where in the previous section it sufficed to ensure non-zero overlap at the minimum value of D, an additional problem surfaces in this case.

Suppose in our limited precision comparisons in the selection function we use the β most significant digits of D and $\hat{P}_j$. Since we wish to output the first quotient digit after the first δ digits of D have arrived, clearly $\beta \le \delta$. The error Δd between the real value of D and the limited precision (β-digit) version of D (denoted $\hat{D}$) which we use for comparison is

$$|\Delta d| \le \sum_{i=\beta+1}^{m} 2r^{-i}$$

since for $i = 1,2,\ldots,m$, $d_i \le 2$. It can be easily shown that

$$|\Delta d| \le \frac{2}{3}\, r^{-\beta}$$

A similar analysis shows that Δp, the error between the real value of $\hat{P}_j$ and the limited precision (β-digit) version of $\hat{P}_j$ (denoted R_j) used for comparison also has the form

$$|\Delta p| \le \frac{2}{3}\, r^{-\beta}$$

Due to the errors Δd and Δp, we see that for any value D_0 of D, the corresponding point on a step of the selection function really represents a height $2\Delta p$ and width $2\Delta d$ centered at this point, and this rectangle must lie completely within the overlap region to guarantee correct quotient digit selection (See Figure 4). With this criterion in mind, we can compute the proper value of δ to ensure sufficient overlap.

Recall $\delta = \beta + 1$ and $K = 2/3$ is the degree of redundancy. Clearly the worst case occurs at the minimum value of D, in this case 1/12. Trivedi [1] has shown that if β and δ satisfy

$$\left(2K + K^2\right) r^{-\delta+2} + K\left(\rho-K\right) r^{-\beta+1} \le \left(1-K\right)\left(K-\frac{1}{2}\right)$$

then sufficient overlap will result. From the above, it can easily be shown that $\beta \ge 4$, and hence $\delta \ge 5$. Now we can compute our selection steps, as depicted in Figure 5. These steps define the selection function below.

D	$4\hat{P}_j$	q_{j+1}
	$\frac{65}{512} \le 4\hat{P}_j$	2
	$\frac{24}{512} \le 4\hat{P}_j \le \frac{65}{512}$	1
$\frac{42}{512} \le D \le \frac{45}{512}$	$-\frac{24}{512} \le 4\hat{P}_j \le \frac{24}{512}$	0
	$-\frac{65}{512} \le 4\hat{P}_j \le -\frac{24}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{65}{512}$	$\bar{2}$
	$\frac{69}{512} \le 4\hat{P}_j$	2
	$\frac{25}{512} \le 4\hat{P}_j \le \frac{69}{512}$	1
$\frac{45}{512} \le D \le \frac{47}{512}$	$-\frac{25}{512} \le 4\hat{P}_j \le \frac{25}{512}$	0
	$-\frac{69}{512} \le 4\hat{P}_j \le -\frac{25}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{69}{512}$	$\bar{2}$
	$\frac{74}{512} \le 4\hat{P}_j$	2
	$\frac{27}{512} \le 4\hat{P}_j \le \frac{74}{512}$	1
$\frac{47}{512} \le D \le \frac{51}{512}$	$-\frac{27}{512} \le 4\hat{P}_j \le \frac{27}{512}$	0
	$-\frac{74}{512} \le 4\hat{P}_j \le -\frac{27}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{74}{512}$	$\bar{2}$
	$\frac{80}{512} \le 4\hat{P}_j$	2
	$\frac{29}{512} \le 4\hat{P}_j \le \frac{80}{512}$	1
$\frac{51}{512} \le D \le \frac{56}{512}$	$-\frac{29}{512} \le 4\hat{P}_j \le \frac{29}{512}$	0
	$-\frac{80}{512} \le 4\hat{P}_j \le -\frac{29}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{80}{512}$	$\bar{2}$
	$\frac{87}{512} \le 4\hat{P}_j$	2
	$\frac{32}{512} \le 4\hat{P}_j \le \frac{80}{512}$	1
$\frac{56}{512} \le D \le \frac{61}{512}$	$-\frac{32}{512} \le 4\hat{P}_j \le \frac{32}{512}$	0
	$-\frac{80}{512} \le 4\hat{P}_j \le -\frac{32}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{80}{512}$	$\bar{2}$

	$\frac{97}{512} \le 4\hat{P}_j$	2
	$\frac{36}{512} \le 4\hat{P}_j \le \frac{97}{512}$	1
$\frac{61}{512} \le D \le \frac{68}{512}$	$-\frac{36}{512} \le 4\hat{P}_j \le \frac{36}{512}$	0
	$-\frac{97}{512} \le 4\hat{P}_j \le -\frac{36}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{97}{512}$	$\bar{2}$
	$\frac{109}{512} \le 4\hat{P}_j$	2
	$\frac{41}{512} \le 4\hat{P}_j \le \frac{109}{512}$	1
$\frac{68}{512} \le D \le \frac{77}{512}$	$-\frac{41}{512} \le 4\hat{P}_j \le \frac{41}{512}$	0
	$-\frac{109}{512} \le 4\hat{P}_j \le -\frac{41}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{109}{512}$	$\bar{2}$
	$\frac{123}{512} \le 4\hat{P}_j$	2
	$\frac{47}{512} \le 4\hat{P}_j \le \frac{123}{512}$	1
$\frac{77}{512} \le D \le \frac{88}{512}$	$-\frac{47}{512} \le 4\hat{P}_j \le \frac{47}{512}$	0
	$-\frac{123}{512} \le 4\hat{P}_j \le -\frac{47}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{123}{512}$	$\bar{2}$
	$\frac{142}{512} \le 4\hat{P}_j$	2
	$\frac{54}{512} \le 4\hat{P}_j \le \frac{142}{512}$	1
$\frac{88}{512} \le D \le \frac{102}{512}$	$-\frac{54}{512} \le 4\hat{P}_j \le \frac{54}{512}$	0
	$-\frac{142}{512} \le 4\hat{P}_j \le -\frac{54}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{142}{512}$	$\bar{2}$
	$\frac{165}{512} \le 4\hat{P}_j$	2
	$\frac{64}{512} \le 4\hat{P}_j \le \frac{165}{512}$	1
$\frac{102}{512} \le D \le \frac{120}{512}$	$-\frac{64}{512} \le 4\hat{P}_j \le \frac{64}{512}$	0
	$-\frac{165}{512} \le 4\hat{P}_j \le -\frac{64}{512}$	$\bar{1}$
	$\le 4\hat{P}_j \le -\frac{165}{512}$	$\bar{2}$
	$\frac{194}{512} \le 4\hat{P}_j$	2
	$\frac{75}{512} \le 4\hat{P}_j \le \frac{194}{512}$	1
$\frac{120}{512} \le D \le \frac{141}{512}$	$-\frac{75}{512} \le 4\hat{P}_j \le \frac{75}{512}$	0
	$-\frac{194}{512} \le 4\hat{P}_j \le -\frac{75}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{195}{512}$	$\bar{2}$
	$\frac{230}{512} \le 4\hat{P}_j$	2
	$\frac{90}{512} \le 4\hat{P}_j \le \frac{230}{512}$	1
$\frac{141}{512} \le D \le \frac{169}{512}$	$-\frac{90}{512} \le 4\hat{P}_j \le \frac{90}{512}$	0
	$-\frac{230}{512} \le 4\hat{P}_j \le -\frac{90}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{230}{512}$	$\bar{2}$
	$\frac{276}{512} \le 4\hat{P}_j$	2
	$\frac{108}{512} \le 4\hat{P}_j \le \frac{276}{512}$	1
$\frac{169}{512} \le D \le \frac{203}{512}$	$-\frac{108}{512} \le 4\hat{P}_j \le \frac{108}{512}$	0
	$-\frac{276}{512} \le 4\hat{P}_j \le -\frac{108}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{276}{512}$	$\bar{2}$
	$\frac{332}{512} \le 4\hat{P}_j$	2
	$\frac{130}{512} \le 4\hat{P}_j \le \frac{332}{512}$	1
$\frac{203}{512} \le D \le \frac{245}{512}$	$-\frac{130}{512} \le 4\hat{P}_j \le \frac{130}{512}$	0
	$-\frac{332}{512} \le 4\hat{P}_j \le -\frac{130}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{332}{512}$	$\bar{2}$
	$\frac{403}{512} \le 4\hat{P}_j$	2
	$\frac{159}{512} \le 4\hat{P}_j \le \frac{403}{512}$	1
$\frac{245}{512} \le D \le \frac{298}{512}$	$-\frac{159}{512} \le 4\hat{P}_j \le \frac{159}{512}$	0
	$-\frac{403}{512} \le 4\hat{P}_j \le -\frac{159}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{403}{512}$	$\bar{2}$

D range	$4\hat{P}_j$ range	q_{j+1}
	$\frac{491}{512} \le 4\hat{P}_j$	2
	$\frac{194}{512} \le 4\hat{P}_j \le \frac{491}{512}$	1
$\frac{298}{512} \le D \le \frac{364}{512}$	$-\frac{194}{512} \le 4\hat{P}_j \le \frac{194}{512}$	0
	$-\frac{491}{512} \le 4\hat{P}_j \le -\frac{194}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{491}{512}$	$\bar{2}$
	$\frac{602}{512} \le 4\hat{P}_j$	2
	$\frac{238}{512} \le 4\hat{P}_j \le \frac{602}{512}$	1
$\frac{364}{512} \le D \le \frac{447}{512}$	$-\frac{238}{512} \le 4\hat{P}_j \le \frac{238}{512}$	0
	$-\frac{602}{512} \le 4\hat{P}_j \le -\frac{238}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{602}{512}$	$\bar{2}$
	$\frac{740}{512} \le 4\hat{P}_j$	2
	$\frac{294}{512} \le 4\hat{P}_j \le \frac{740}{512}$	1
$\frac{447}{512} \le D \le 1$	$-\frac{294}{512} \le 4\hat{P}_j \le \frac{294}{512}$	0
	$-\frac{740}{512} \le 4\hat{P}_j \le -\frac{294}{512}$	$\bar{1}$
	$4\hat{P}_j \le -\frac{740}{512}$	$\bar{2}$

To conclude this section, we present an example of radix 4 on-line division with redundant operands.

N: -1 -1 -2 0 0 1 -1 -1 0 1 1 0 -1 0 0 1

D: 2 0 1 2 0 1 0 1 -1 -1 1 1 -1 -2 2 -1

$\hat{P}_j$	q_{j+1}
¯0.34375	¯2
¯0.3266601563	¯2
¯0.2602539063	¯2
0.006011962891	0
0.02340698242	0
0.09396362305	1
¯0.1462204456	¯1
¯0.06053872274	0
¯0.2438121438	¯2
0.07085485756	0
0.2847008258	2
0.09175475128	1
¯0.1566750032	¯1
¯0.1030076046	¯1
0.1116619899	1
¯0.07701444858	0

6. Conclusions

We have presented two algorithms for radix 4 on-line division, along with a proof of their correctness. The second algorithm (radix 4 on-line division with redundant operands) is of particular interest because it possesses two advantages. First, it has the property of closure, that is, both the operands and the quotient have redundant representation. Any machine implementing such a division algorithm could be based entirely on redundant representation of numbers, and costly conversion from redundant to non-redundant format could be avoided (except, of course, before an output operation). Secondly, the redundancy of the dividend, divisor, and quotient digits means that the partial remainders $\hat{P}_j$ are also redundant. This means that the basic recursions of Step 2.2 of the algorithm can be computed using carry-free addition. On the other hand, the selection function for this algorithm is more complex than either the first algorithm (radix 4 on-line division with non-redundant operands) or the algorithm for binary on-line division with non-redundant operands [1].

Topics which remain to be investigated in this area include (a) comparison of the performance of on-line division versus conventional division algorithms, (b) the development of on-line algorithms to compute the trigonometric, nth root, nth power, log and exponential functions, and (c) the study of new arithmetic unit architectures to exploit such on-line algorithms.

7. References

[1] K. S. Trivedi and M. D. Ercegovac, "On-line algorithms for division and multiplication," IEEE Trans. on Computers, vol C-26, no. 7, July 1977.

[2] J. E. Robertson, "A new class of digital division methods," IRE Trans. Electron. Comput. vol EC-7, pp. 218-222, Sept. 1958.

[3] A. Avizienis, "Signed digit number representation for fast parallel arithmetic," IRE Trans. Electron. Comput., vol. EC-10, pp. 389-400, 1961.

[4] J. O. Campeau, "Communications and sequential problems in the parallel processor," in Parallel Processor Systems, Technologies, and Applications. New York: Spartan, 1970.

[5] N. Metropolis, and R. L. Ashenhurst, "Significant Digit Computer Arithmetic," IRE Trans. Electron. Comput., vol. EC-7, pp. 265-267, 1978.

[6] R. M. Russell, "The Cray-1 Computer System," Comm. ACM, vol. 21, no. 1, pp. 63-72, 1978.

[7] T. C. Chen, "Overlap and Pipeline Processing," in Introduction to Computer Architecture, H. S. Stone (ed.), Science Research Associates, Chicago, 1975.

[8] M. J. Irwin, "An arithmetic unit for on-line computation," Ph. D. dissertation, Dept. of Comp. Sci., Univ. of Illinois, Urbana-Champaign, IL, 1977.

[9] M. D. Ercegovac, "A general hardware-oriented method for evaluation of functions and computations on a digital computer," IEEE Trans. on Computers, vol. C-26, no. 7, July 1977.

[10] M. D. Ercegovac, "An on-line square rooting algorithm," this Symposium.

[11] S. K. Nandi and E. V. Krishnamoorthy, "A simple technique for digital Division," CACM 10, pp. 299-301, May 1969.

[12] E. V. Krishnamoorthy, "On a range transformation technique for Division," IEEE Trans. Comput. C-19, 2(Feb. 1970), pp. 157-160.

[13] C. Tung, "A division algorithm for signed digit arithmetic," IEEE Trans. Comput. C-17, 9 (Sept. 1968), pp. 887-889.

[14] D. E. Atkins, Ph. D. dissertation, Dept. of Comp. Sci., Univ. of Ill. Urbana, June 1970.

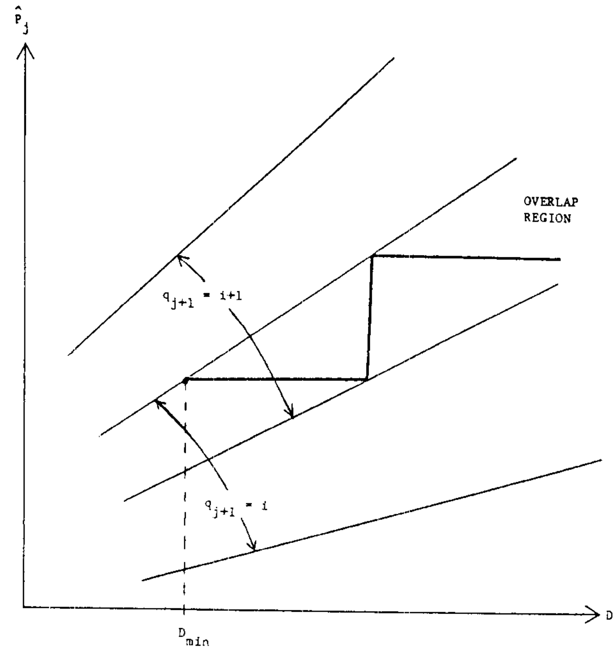

Figure 2: Computing selection steps for radix 4 on-line division with non-redundant operands

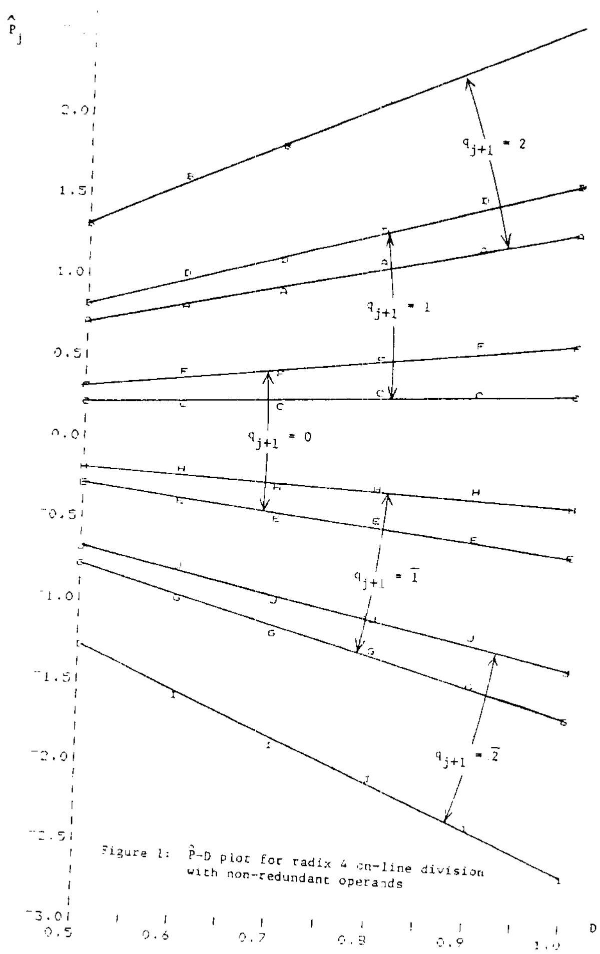

Figure 1: $\hat{P}$-D plot for radix 4 on-line division with non-redundant operands

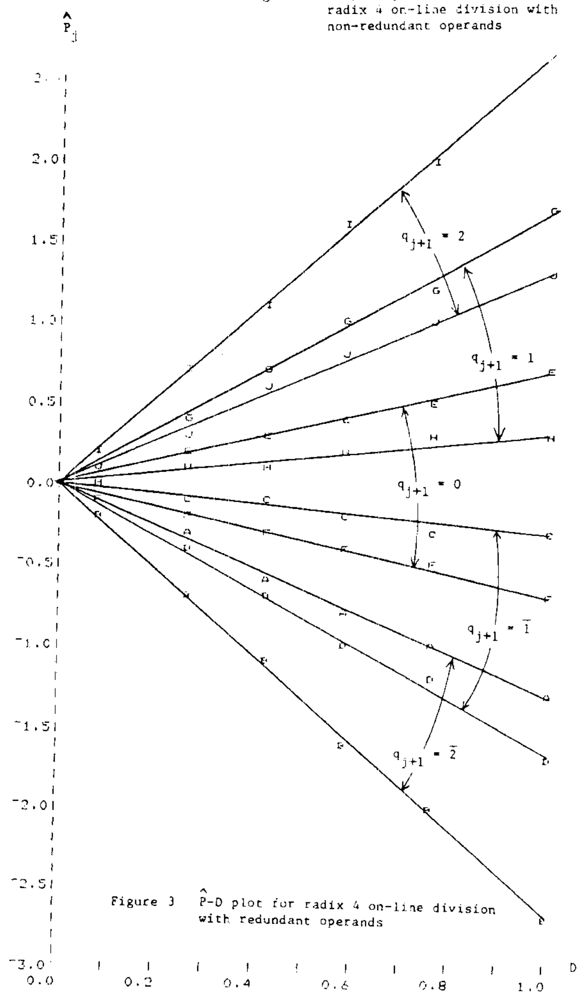

Figure 3 $\hat{P}$-D plot for radix 4 on-line division with redundant operands

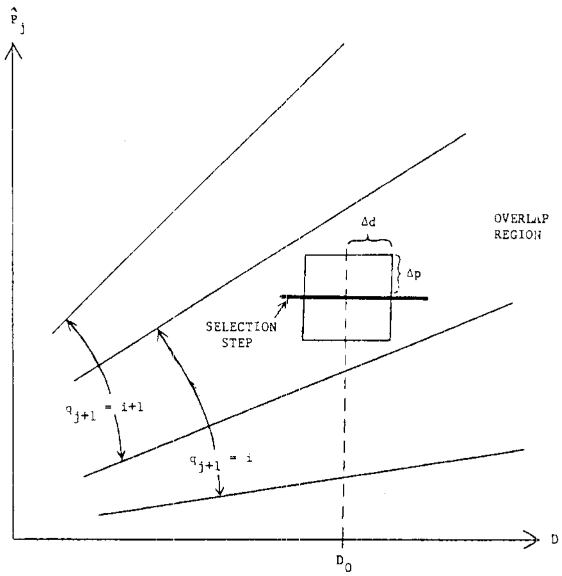

Figure 4

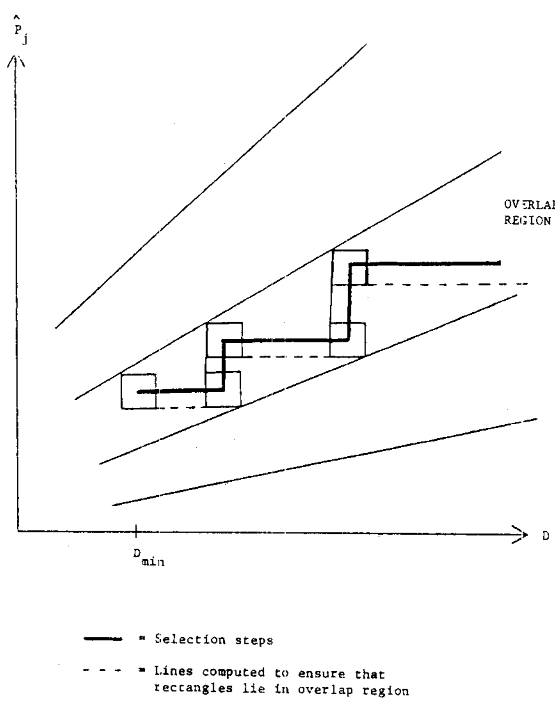

Figure 5: Computing selection steps for radix 4 on-line division with redundant operands