L-141

Arithmetic in a One's-Complement Computer.

A. Introduction.

The main parts of the arithmetic section of a one's-complement computer are the X-Register (abbreviated XR), the Q-Register, QR, and the Accumulator, AC. Numbers may be taken from the cells of the memory section and put into these registers and the four fundamental arithmetic operations performed on them there. For each of the operations we shall indicate the process through which the machine goes and prove that this process gives the correct result.

The computer operates with 24 digit signless binary numbers, each of which represents a positive or negative 23 digit binary number (perhaps having superfluous zeros.) The numbers with which the machine operates will be referred to as digital numbers, and the numbers for which these stand as binary numbers, or simply numbers.

It is first necessary to investigate the way in which a number is represented in the computer.

B. The Correspondence.

Consider the signless binary numbers (digital numbers) of 24 digits, $X = x_{23} \ldots x_0$, $x_i = 0$ or 1, excluding that digital number consisting exclusively of 1's. The <u>one's-complement</u> of such a digital number X is defined to be $X' = (2^{24} - 1) - X$.

1/27

Since $2^{24} - 1 = 11 \ldots 11$, it is seen that $X' = x'_{23} \ldots x'_0$ where $x'_i = \left\{ \begin{matrix} 1 \\ 0 \end{matrix} \right\}$ according as $x_i = \left\{ \begin{matrix} 0 \\ 1 \end{matrix} \right\}$. Note that $(x'_i)' = x_i$, $(X')' = X$.

Between the above set of digital numbers and the set of signed binaries of 23 digits x (perhaps with superfluous zeros), $1 - 2^{23} \leq x \leq 2^{23} - 1$, set up a one-one correspondence as follows:

If $x \geq 0$ and $|x| = x_{22} \ldots x_0$ then let $x \longleftrightarrow X = 0x_{22} \ldots x_0$. If $x < 0$ and $|x| = x_{22} \ldots x_0$ then let $x \longleftrightarrow X = 1x'_{22} \ldots x'_0 = 2^{24} - 1 - |x|$, i.e., if x is negative let it correspond to the one's-complement of the image of its absolute value. Note that the two sets between which the correspondence is defined are actually cardinally equivalent.

The XR, QR and memory cells represent binary numbers x, $1 - 2^{23} \leq x \leq 2^{23} - 1$, as digital numbers by means of the above correspondence. The AC, however, can represent a number a, $1 - 2^{47} \leq a \leq 2^{47} - 1$, as a 48 digit digital number by virtue of an analogous one-one correspondence. If A is one of these 48 digit digital numbers (not consisting exclusively of 1's) then its one's-complement is by definition the digital number $A' = (2^{48} - 1) - A = a'_{47} \ldots a'_0$ where

$a'_j = \binom{1}{0}$ according as $a_j = \binom{0}{1}$. Again, $(a'_j)' = a_j$ and $(A')' = A$. The analogous one-one correspondence by virtue of which the AC holds any signed 47 digit number is the following one. Let $|a| = a_{46} \ldots a_0$. If $a \geq 0$ then $a \longleftrightarrow A = 0a_{46} \ldots a_0$. If $a < 0$ then $a \longleftrightarrow A = 1a'_{46} \ldots a'_0 = 2^{48} - 1 - |a| = 2^{48} - 1 + a$.

Let $X_A = x_{23} \ldots x_{23} x_{23} x_{22} \ldots x_1 x_0$ if $X = x_{23} \ldots x_0$. We have the easily verified

Lemma I: If X is the representation of x on XR, then X_A is the representation of x on AC, i.e., if $x \longleftrightarrow X$ then $x \longleftrightarrow X_A$.

C. Addition.

A digital number $X = x_{23} \ldots x_0$ in XR is "added" to a digital number $A = a_{47} \ldots a_0$ in AC in the following way. The machine computes

$$A = a_{47} a_{46} \ldots a_{25} a_{24} a_{23} a_{22} \ldots a_0$$

$$\text{minus } X'_A = x'_{23} x'_{23} \ldots x'_{23} x'_{23} x'_{23} x'_{22} \ldots x'_0$$

and if there is a borrow left over at the left-hand end, this borrow is subtracted from the resulting difference to give a result called the "sum" of A and X, or $A \oplus X$.

The subtraction employed is known as the end-around borrow method. The subtraction of the borrow from the first difference may generate another series of borrows, but this second series cannot run for more than 48 places, since in the first subtraction a borrow can originate only at a configuration of the form $\frac{\substack{0\\1}}{1}$, and the 1 in the first difference will kill any borrow generated in the second subtraction.

It must be shown that, within the range of the computer, if $x \longleftrightarrow X$ and $a \longleftrightarrow A$ then $a + x \longleftrightarrow A \ominus X$. Moreover, it will be shown that if the result runs over the capacity of the machine it is still correct modulo $2^{48} - 1$.

If $X_A' \leq A$ then no end-around borrow can be generated and $A \ominus X = A - X_A'$. End around borrow will occur if and only if $X_A' > A$. Using end-around borrow is equivalent to adding 2^{48} to A, subtracting X_A' from their sum, and then subtracting 1 from this difference. Thus if $X_A' > A$, $A \ominus X = ((A + 2^{48}) - X_A') - 1 = A + X_A$. Concisely,

$$\text{if } X_A' \leq A \text{ then } A \ominus X = A - X_A',$$

$$\text{if } X_A' > A \text{ then } A \ominus X = A + X_A.$$

Let x, $1 - 2^{23} \leq x \leq 2^{23} - 1$, $|x| = x_{22} \ldots x_0$, be a number represented on XR by X, and

L-141

a, $1 - 2^{47} \leq a \leq 2^{47} - 1$, $a = a_{46} \ldots a_0$ be a number represented on AC by A. Notice first that

$0 \leq |a| + |x| \leq (2^{47} - 1) - (2^{23} - 1)$ and

$0 \leq |a| - |x| \leq 2^{47} - 1$. There are four cases, one for each of the possible distributions of signs on x and a, which arise in the addition of x to a by means of their representatives on the computer. Each case must be considered separately in order to prove that in general $a + x \longleftrightarrow A \oplus X$.

Case I: $x \geq 0$ and $a \geq 0$. In this case

$a \longleftrightarrow A = 0a_{46} \ldots a_0 = |a|$

$x \longleftrightarrow X = 0x_{22} \ldots x_0 = |x|$

$X_A = 0 \ldots 0x_{22} \ldots x_0 = |x|$ and

$X_A' = 1 \ldots 1\, x_{22}' \ldots x_0' = (2^{48} - 1) - X_A = (2^{48} - 1) - |x|$.

Since $X_A' > A$, $A \oplus X = A + X_A = |a| + |x|$. If $|a| + |x| \leq 2^{47} - 1$ then $A \oplus X \longleftrightarrow |a| + |x| = a + x$.

But if $|a| + |x| > 2^{47} - 1$, then, since $|a| + |x| \leq (2^{47} - 1) + (2^{23} - 1)$, $|a| + |x| = (2^{47} - 1) + n$ where $0 < n \leq 2^{23} - 1$. Then since $0 < 2^{47} - n \leq 2^{47} - 1$, we have $A \oplus X = (2^{48} - 1) - (2^{47} - n) \longleftrightarrow -(2^{47} - n) = m$.

But $(|a| + |x|) - m = 2^{48} - 1$ so $a + x = |a| + |x| \equiv m \pmod{2^{48} - 1}$ and

$A \oplus X \longleftrightarrow m \equiv a + x \pmod{2^{48} - 1}$.

L-141

<u>Case II</u>: $x < 0$ and $a \geq 0$. In this case

$a \longleftrightarrow A = 0a_{46} \dots a_0 = |a|$

$x \longleftrightarrow X = 1x'_{22} \dots x'_0$

$X_A = 1 \dots 1\, x'_{22} \dots x'_0 = (2^{48} - 1) - |x|$ and

$X'_A = 0 \dots 0\, x_{22} \dots x_0 = |x|$.

If $X'_A \leq A$ then $A \oplus X = A - X'_A = |a| - |x|$.

But if $X'_A \leq A$ then $|x| \leq |a|$ and $0 \leq |a| - |x| \leq 2^{47} - 1$

and $A \oplus X \longleftrightarrow |a| - |x| = a + x$. On the other hand, if

$X'_A > A$ then $A \oplus X = A + X_A = |a| + (2^{48} - 1) - |x| =$

$(2^{48} - 1) - (|x| - |a|)$. Also, if $X'_A > A$ then

$|x| > |a|$ and $0 < |x| - |a| \leq 2^{47} - 1$ and

$A \oplus X \longleftrightarrow -(|x| - |a|) = a + x$.

<u>Case III</u>: $x \geq 0$ and $a < 0$. In this case

$x \longleftrightarrow X = 0x_{22} \dots x_0 = |x|$

$X_A = 0 \dots 0\, x_{22} \dots x_0 = |x|$

$X'_A = 1 \dots 1\, x'_{22} \dots x'_0 = (2^{48} - 1) - X_A = (2^{48} - 1) - |x|$,

$a \longleftrightarrow A = 1a'_{46} \dots a'_0 = (2^{48} - 1) - |a|$.

If $X'_A \leq A$ then $A \oplus X = A - X'_A =$

$(2^{48} - 1) - |a| - ((2^{48} - 1) - |x|) = -|a| + |x|$. If

$X'_A \leq A$ then $|x| \geq |a|$ so $0 \leq -|a| + |x| \leq 2^{47} - 1$

and $A \oplus X \longleftrightarrow -|a| + |x| = a + x$.

If $X_A' > A$ then $A \oplus X = A + X_A' = (2^{48} - 1) - (|a| - |x|)$.

Since $X_A' > A$, $|x| < |a|$ and $0 < |a| - |x| \leq 2^{47} - 1$ so

$A \oplus X \longleftrightarrow -(|a| - |x|) = a + x$.

<u>Case IV</u>: $x < 0$ and $a < 0$. In this case

$a \longleftrightarrow A = 1a_{46}' \ldots a_0' = (2^{48} - 1) - |a|$

$x \longleftrightarrow X = 1x_{22}' \ldots x_0$

$X_A = 1 \ldots 1\, x_{22}' \ldots x_0$ and

$X_A' = 0 \ldots 0\, x_{22} \ldots x_0 = |x|$.

Since $X_A' < A$, $A \oplus X = A - X_A' = (2^{48} - 1) - (|a| + |x|)$.

If $|a| + |x| \leq 2^{47} - 1$ then $A \oplus X \longleftrightarrow -(|a| + |x|) = a + x$.

But if $|a| + |x| > 2^{47} - 1$ then, since
$|a| + |x| \leq (2^{47} - 1) + (2^{23} - 1)$, $|a| + |x| = (2^{47} - 1) + n$

where $0 < n \leq 2^{23} - 1$. Then $(2^{48} - 1) - (|a| + |x|) =$

$2^{47} - n$, and since $0 < 2^{47} - n \leq 2^{47} - 1$,

$A \oplus X \longleftrightarrow 2^{47} - n = m$. But $m - (-(|a| + |x|)) =$

$2^{48} - 1$ so $a + x = -(|a| + |x|) \equiv m \pmod{2^{48} - 1}$

and $A \oplus X \longleftrightarrow m \equiv a + x \pmod{2^{48} - 1}$.

The above arguments prove that if x, a and $a + x$ are within the ranges of XR, AC and AC respectively, then $a + x \longleftrightarrow A \oplus X$ and if $a + x$ is beyond the range of AC

L-141

the $A \oplus X$ is still the correct image of $a + x$ modulo $2^{48} - 1$.

D. Subtraction.

Subtraction is performed by the computer by subtracting X_A from A using end-around borrow when necessary, i.e., $A \ominus X = A \oplus X'$. If $a \longleftrightarrow A$ and $x \longleftrightarrow X$ then $-x \longleftrightarrow X'$ and $A \oplus X \longleftrightarrow a + x$ or $A \oplus X \longleftrightarrow m \equiv a + x \pmod{2^{48} - 1}$ and $A \ominus X = A \ominus X' \longleftrightarrow a + (-x) = a - x$ or $A \ominus X \longleftrightarrow n \equiv a - x \pmod{2^{48} - 1}$ as the case may be. Thus the desired correspondence is preserved by machine subtraction.

E. Basic Arithmetic Operations.

Below is a table summarizing the basic arithmetic operations.

NUMBERS SUBTRACTED INTO AC IN THE BASIC OPERATIONS

OPERATION	AC DIGITS $a_{47}\ a_{46} \ldots a_{24}\ a_{23}\ a_{22} \ldots a_1\ a_0$
ADD XR TO AC	$x'_{23}\ x'_{23} \ldots x'_{23}\ x'_{23}\ x'_{22} \ldots x'_1\ x'_0$
SUBTRACT XR FROM AC	$x_{23}\ x_{23} \ldots x_{23}\ x_{23}\ x_{22} \ldots x_1\ x_0$
ABSOLUTE ADD XR TO AC	
IF $x_{23} = 0$	$1\ 1 \ldots 1\ 1\ x'_{22} \ldots x'_1\ x'_0$
IF $x_{23} = 1$	$1\ 1 \ldots 1\ 1\ x_{22} \ldots x_1\ x_0$
ABSOLUTE SUBTRACT XR FROM AC	
IF $x_{23} = 0$	$0\ 0 \ldots 0\ 0\ x_{22} \ldots x_1\ x_0$
IF $x_{23} = 1$	$0\ 0 \ldots 0\ 0\ x'_{22} \ldots x'_1\ x'_0$

8/27

L-141

The only operations in this table which have not been proved to yield correct results are the operations of absolute add and absolute subtract. From a consideration of the digits put into AC in these operations it is easily verified that they correspond to the true operations of addition and subtraction of absolute values.

F. Multiplication.

In addition to XR and AC the arithmetic section contains a Q-Register, QR, which holds a 23 digit signed binary as a digital number of 24 digits just as XR does. All the digits of QR or of AC may be shifted left any number of places from 1 to 47. The shift is circular, i.e., a digit shifting out the left end of the register is not lost but shifts to the right end.

In the proof of the multiplication algorithm we will need the following two lemmas.

Lemma II: Let $A = a_{47} \dots a_0$ be a digital number in AC. Let $\bar{A}$ be the result of shifting A circularly to the left s places where $s < 48$. Then $\bar{A} \equiv A2^s \pmod{2^{48} - 1}$, where $A2^s$ is the product of 2^s with the digital number A (not its image).

Proof:

$$A2^s = a_{47} \dots a_{48-s} \overbrace{a_{47-s} \dots a_0 \underbrace{0\,0 \dots 0\,0}_{s \text{ places}}}^{48 \text{ places}} \quad \text{and}$$

9/27

$\bar{A} = a_{47-s} \cdots a_o \, a_{47} \cdots a_{48-s}$. But

$a_{47} \cdots a_{48-s} \underbrace{0\,0 \ldots 0\,0}_{48 \text{ places}} = 2^{48} (a_{47} \cdots a_{48-s})$ and therefore

$A2^s - (2^{48} - 1)(a_{47} \cdots a_{48-s}) = a_{47-s} \cdots a_o \, a_{47} \cdots a_{48-s} = \bar{A}$

and $\bar{A} \equiv A2^s \pmod{2^{48} - 1}$.

Lemma III: Let A and B be two 48 place digital numbers and let $A \boxminus B$ denote their difference using end-around borrow if necessary. Let $\bar{A}, \bar{B}, \overline{A \boxminus B}$ denote the results of shifting each s places where $s < 48$. Then $\bar{A} \boxminus \bar{B} = \overline{A \boxminus B}$.

Proof: All congruences here are modulo $2^{48} - 1$. It is implicit from the result on page 4 that $A \boxminus B \equiv A - B$, $\bar{A} \boxminus \bar{B} \equiv \bar{A} - \bar{B}$. From Lemma II, $\bar{A} \equiv 2^s A$, $\bar{B} \equiv 2^s B$, and hence $\bar{A} - \bar{B} \equiv 2^s(A - B)$. Thus $\bar{A} \boxminus \bar{B} \equiv 2^s(A \boxminus B) \equiv \overline{(A \boxminus B)}$. But $0 \leq \bar{A} \boxminus \bar{B} \leq 2^{48} - 1$ and $0 \leq \overline{A \boxminus B} \leq 2^{48} - 1$ and therefore $\bar{A} \boxminus \bar{B} = \overline{A \boxminus B}$.

When the computer adds and when it subtracts it computes an end-around borrow difference, which operation we have denoted by $\boxminus$ in Lemma III. Thus Lemma III holds for both machine addition and machine subtraction.

Lemma IV: Let the AC digital number $A \longleftrightarrow a$ where $|a| \leq 2^{23} - 1$. Then if $s \leq 24$, $\bar{A} \longleftrightarrow a2^s$.

The truth of this lemma is most easily seen from a consideration of cases $a_{23} = 0$ and $a_{23} = 1$.

We now proceed to the multiplication algorithm proper.

L-141

Multiplication Algorithm

Initial contents

QR contains multiplier
XR contains multiplicand
AC contains number to which the product is to be added.

Algorithm

1. Shift AC 24 places left.

2. If $q_{23} = 1$, subtract XR into AC.

   If $q_{23} = 0$, proceed directly to step 3.

3. Do 24 times:

   a. Shift AC one place left.

   b. If $q_{23} = 1$ add XR into AC, otherwise proceed to 3C.

   c. Shift QR one place left.

4. If $q_{23} = 1$, add XR into AC.

Result

AC contains produce plus its initial contents.
QR contains multiplier.

Let $a \longleftrightarrow A = a_{47} \ldots a_0$, $x \longleftrightarrow X = x_{23} \ldots x_0$ and $q \longleftrightarrow Q = q_{23} \ldots q_0$. At the end of the operation AC has been shifted left 48 places and is back where it started. In view of Lemma III we may consider the AC held stationary and XR shifted along it each step. Since QR is shifted each step we

L-141

may consider the machine to examine the proper digit of Q with QR held stationary. Examination of the algorithm in the light of these remarks shows that the machine computes the sum

$$
\begin{array}{lcccccccccccccl}
 & a_{47} & a_{46} & a_{45} & \cdots & a_{25} & a_{24} & a_{23} & a_{22} & \cdots & a_1 & a_0 & \leftrightarrow & a \\
\oplus (q_{23}) & x'_{23} & x'_{22} & x'_{21} & \cdots & x'_1 & x'_0 & x'_{23} & x'_{22} & \cdots & x'_{23} & x'_{23} & \leftrightarrow & -x \cdot 2^{24} \cdot q_{23} \\
\oplus (q_{23}) & x_{23} & x_{23} & x_{22} & \cdots & x_1 & x_0 & x_{23} & \cdots & & & x_{23} & \leftrightarrow & x \cdot 2^{23} \cdot q_{23} \\
\oplus (q_{22}) & x_{23} & x_{23} & x_{23} & x_{22} & \cdots & x_1 & x_0 & x_{23} & \cdots & & x_{23} & \leftrightarrow & x \cdot 2^{22} \cdot q_{22} \\
\vdots & & & \vdots & & & \vdots & & & & & \vdots & & \vdots \\
\oplus (q_1) & x_{23} & \cdots & & & x_{23} & x_{23} & x_{22} & \cdots & x_1 & x_0 & x_{23} & \leftrightarrow & x \cdot 2 \cdot q_1 \\
\oplus (q_0) & x_{23} & \cdots & & & x_{23} & x_{23} & x_{23} & x_{22} & \cdots & x_1 & x_0 & \leftrightarrow & x \cdot 2^0 \cdot q_0 \\
\oplus (q_{23}) & x_{23} & \cdots & & & x_{23} & x_{23} & x_{23} & x_{22} & \cdots & x_1 & x_0 & \leftrightarrow & x \cdot 2^0 \cdot q_{23}
\end{array}
$$

12/27

L-141

The complement is used in the second line because machine subtraction is the same as machine addition of the complement. The factors q_i to the left simply indicate whether or not the term is to be included in the total sum (according as $q_i = 1$ or 0). The correspondences to the right follow from Lemma IV. By section C the total machine sum corresponds to the sum of the images or to some number which is congruent modulo $2^{48} - 1$ to the sum of the images. Let P be the machine sum of the digital numbers. Then

$$P \longleftrightarrow p \equiv a - x \cdot 2^{24} \cdot q_{23} + \sum_{i=23}^{0} x \cdot 2^i \cdot q_i + x \cdot q_{23} \pmod{2^{48} - 1}.$$

If $q_{23} = 0$ then $Q \longleftrightarrow q = 0\, q_{22} \cdots q_0 \geq 0$ and

$$a - x\, 2^{24}\, q_{23} + \sum_{i=23}^{0} x\, 2^i\, q_i + x\, q_{23} = a + x \sum_{i=23}^{0} q_i\, 2^i =$$

$a + x\,q$ and $P \longleftrightarrow p \equiv a + q\,x \pmod{2^{48} - 1}$.

If, on the other hand, $q_{23} = 1$, then

$Q \longleftrightarrow q = -0\, q_{22}' \cdots q_0' < 0$ and

$$a - x \cdot 2^{24} \cdot q_{23} + \sum_{i=23}^{0} x \cdot 2^i \cdot q_i + x \cdot q_{23} = a - (x(2^{24}-1) - \sum_{i=23}^{0} x \cdot 2^i \cdot q_i) =$$

$$a - \sum_{i=23}^{0} x \cdot 2^i \cdot (1 - q_i) = a - x \sum_{i=23}^{0} q_i'\, 2^i = a + q\,x.$$ Thus in this case as well as the first

$P \longleftrightarrow p \equiv a + q\,x \pmod{2^{48} - 1}$.

L-141

D. Division.

In performing the division operation the computer goes through the process indicated in the following diagram.

Division Algorithm

Initial Contents

QR is cleared
XR contains divisor
AC contains dividend

Algorithm

1. Case I: If $a_{47} = 0$, go to step 2.

   Case II: If $a_{47} = 1$, absolute subtract XR into AC.

2. Shift AC 24 places.

3. Do 24 times:

   a. Shift AC one place.

   b. If $a_{24} = x_{23}$, subtract XR into AC and insert 1 in q_0.

      If $a_{24} \neq x_{23}$, add XR into AC.

   c. Shift QR one place.

4. Do one of the following:

   a. If $a_{24} = 1$, if case I: Absolute add XR into AC, replace q_0 by q_{23}.

   b. If $a_{24} = 1$, if case II: Absolute add XR into AC.

   c. If $a_{24} = 0$, if case I: No action.

   d. If $a_{24} = 0$, if case II: Replace q_0 by q_{23}.

Final Contents

QR contains quotient
XR contains divisor
AC contains least non-negative remainder.

In the proof of the validity of this algorithm we shall need the

<u>Lemma V</u>: Let $f(x) = 2x - 1$. Then

$$f(x_0)\,2^n + \sum_{k=n-1}^{0} f(x_{k+1})\,2^k = \sum_{k=n}^{0} x_k\,2^k + (x_0 - 1)(2^{n+1} - 1).$$

Proof: $f(x_0)\,2^n + \sum_{k=n-1}^{0} f(x_{k+1})\,2^k =$

$$(2x_0 - 1)\,2^n + \sum_{k=n-1}^{0} (2x_{k+1} - 1)\,2^k =$$

$$\sum_{k=n-1}^{0} x_{k+1}\,2^{k+1} + x_0\,2^{n+1} - 2^n - \sum_{k=n-1}^{0} 2^k =$$

$$\sum_{k=n}^{1} x_k\,2^k + x_0\,2^0 + x_0\,2^{n+1} - 2^{n+1} + 1 - x_0 =$$

$$\sum_{k=n}^{0} x_k\,2^k + (x_0 - 1)(2^{n+1} - 1).$$

<u>Case I</u>. We will first consider the case of two positive numbers. Throughout the proof use will be made of Lemmas II and III. The AC will be considered stationary and the MR slid along it an amount which an examination of the algorithm will make evident. Similarly QR will be considered stationary and the computer will be assumed to insert the proper digit in the proper place. In the case of two non-negative numbers the division algorithm reduces to the following.

L-141

$Q = q_{23} \cdots q_o \longleftrightarrow q$

$X = x_{23} \cdots x_o \longleftrightarrow x = 0\, x_{22} \cdots x_o \geqslant 0, \; x_{23} = 0$

$A = a_{47} \cdots a_o \longleftrightarrow a = 0\, a_{46} \cdots a_o \geqslant 0, \; a_{47} = 0$

1°. Shift AC 25 places
Subtract XR into AC and insert 1 in q_o.
Shift QR one place.

2°. Do 23 times
a. Shift AC one place.

b. If $a_{24} = 0$ subtract XR into AC and insert 1 in q_o.

If $a_{24} = 1$ add XR into AC.

c. Shift QR one place.

3°. Do one of the following:

a. If $a_{24} = 1$: Add XR into AC, replace q_o by q_{23}.

b. If $a_{24} = 0$: No action.

Step 1°. Make $q_o = 1$ and compute

$$\begin{array}{lcl} & a_{47}\, a_{46} \cdots a_{25}\, a_{24}\, a_{23}\, a_{22} \cdots a_0 & \longleftrightarrow a \\ \ominus & 0 \quad x_{23} \cdots\cdots x_o \quad 0 \quad \cdots 0 & \longleftrightarrow x\, 2^{23} \\ \hline = & n^1_{47}\, n^1_{46} \cdots n^1_{25}\, n^1_{24}\, n^1_{23}\, n^1_{22} \cdots n^1_0 = N_1 & \longleftrightarrow a - x2^{23} = n_1. \end{array}$$

If we define $f(t) \equiv 2t - 1$, we have, since $q_o = 1$,

$n_1 = a - f(q_o)\, x\, 2^{23}$. Now let us assume that not only do X and A represent non-negative numbers, but also that $0 < x \leqslant 2^{23} - 1$ and $0 < a < x\, 2^{23}$. Then $-x\, 2^{23} < n_1 < 0$ and $|n_1| < x\, 2^{23}$. But $x\, 2^{23} \leqslant (2^{23} - 1)\, 2^{23} < 2^{46} - 1$, and

$|n_1| < 2^{46} - 1$. Therefore $n^1_{47} = n^1_{46}$ and since $n_1 < 0$, $n^1_{46} = 1$. Since in substep 1 of step 2^0 n^1_{46} is compared (notice that n^1_{46} is now the n_{24} referred to in 2^0) with $x_{23} = 0$, the operation in substep 1 is $\oplus$ and q_{23} is left as 0.

Step 2^0. Substep 1: Compute

$$\begin{array}{l}
\quad n^1_{47}\ n^1_{46}\ n^1_{45}\ \cdots\ n^1_{24}\ n^1_{23}\ n^1_{22}\ \cdots\cdots\ n^1_0 \qquad \leftrightarrow n_1 \\
\oplus\ \underline{0\quad 0\quad x_{23}\ \cdots\cdots\cdots\ x_0\quad 0\ \cdots\ 0} \qquad \leftrightarrow x2^{22} \\
= n^2_{47}\ n^2_{46}\ n^2_{45}\ \cdots\ n^2_{24}\ n^2_{23}\ n^2_{22}\ \cdots\cdots\ n^2_0 = N_2 \leftrightarrow n_1 + x2^{22} = n_2.
\end{array}$$

Since $q_{23} = 0$, $n_2 = n_1 - f(q_{23})\ x\ 2^{22}$. Also $n_2 = n_1 + x\ 2^{22} = n - x\ 2^{22}$. Thus $n_2 < x\ 2^{23} - x\ 2^{22} = x\ 2^{22}$ and $n_2 > - x\ 2^{22}$ or $|n_2| < x\ 2^{22}$. But $x\ 2^{22} < 2^{45} - 1$ so $|n_2| < 2^{45} - 1$. Therefore $n^2_{47} = n^2_{46} = n^2_{45}$ and since the operation of substep 2 is $\ominus$ or $\oplus$ according as $n^2_{45} = 0$ or 1, it is $\ominus$ or $\oplus$ according as $n_2 > 0$ or $n_2 < 0$.

Substep 2. Compute

$$\begin{array}{l}
\qquad n^2_{47}\ n^2_{46}\ n^2_{45}\ n^2_{44}\ \cdots\ n^2_{24}\ n^2_{23}\ n^2_{22}\ n^2_{21}\ \cdots\ n^2_0 \qquad \leftrightarrow n_2 \\
\ominus \text{ or } \oplus\ \underline{0\quad 0\quad 0\quad x_{23}\ \cdots\cdots\cdots\ x_0\quad 0\ \cdots\ 0} \qquad \leftrightarrow x\ 2^{21} \\
= \quad n^3_{47}\ n^3_{46}\ n^3_{45}\ n^3_{44}\ \cdots\ n^3_{24}\ n^3_{23}\ n^3_{22}\ n^3_{21}\ \cdots\ n^3_0 = N_3 \leftrightarrow n_2 \mp x2^{21} = n_3
\end{array}$$

Since q_{22} is 1 or 0 according as the operation is $\ominus$ or $\oplus$,

L-141

$n_3 = n_2 - f(q_{22}) \times 2^{21}$. Also $n_3 = n_2 \mp x\, 2^{21}$ according as $n_2 < 0$ or $n_2 \geq 0$. Thus

$|n_3| = |x\, 2^{21} - |n_2||$, and since $x\, 2^{21} - |n_2| \leq x\, 2^{21}$ and $|n_2| - x\, 2^{21} < x\, 2^{22} - x\, 2^{21} = x\, 2^{21}$, we have $|n_3| \leq x\, 2^{21}$. But $x\, 2^{21} < 2^{44} - 1$ and so $|n_3| < 2^{44} - 1$. Therefore $n^3_{47} = n^3_{46} = n^3_{45} = n^3_{44}$, and since the operation of substep 3 is $\ominus$ or $\oplus$ according as $n^3_{44} = 0$ or 1 it is $\ominus$ or $\oplus$ according as $n_3 \geq 0$ or $n_3 < 0$.

For a general k, $1 < k < 23$, assume that $|n_k| \leq x\, 2^{24-k} < 2^{47-k} - 1$. Then $n^k_{47} = n^k_{46} = \cdots = n^k_{47-k}$. Since the next operation is $\ominus$ or $\oplus$ according as $n^k_{47-k} = 0$ or 1, it is $\ominus$ or $\oplus$ according as $n_k \geq 0$ or $n_k < 0$.

Substep k: Compute

$$
\begin{array}{lcccccccccl}
 & \cdot n^k_{47} & \cdots & n^k_{47-k} & n^k_{46-k} & \cdots & n^k_{23-k} & n^k_{22-k} & \cdots & n^k_0 & \leftrightarrow n_k \\
\ominus \text{ or } \oplus & 0 & \cdots & 0 & x_{23} & \cdots & x_0 & 0 & \cdots & 0 & \leftrightarrow x2^{23-k} \\
\hline
= & n^{k+1}_{47} & \cdots & n^{k+1}_{47-k} & n^{k+1}_{46-k} & \cdots & n^{k+1}_{23-k} & n^{k+1}_{22-k} & \cdots & n^{k+1}_0 = n_{k+1} & \leftrightarrow \\
 & & & & & & & & & & n_k \mp x2^{23-k} = n_{k+1}
\end{array}
$$

Since $q_{24-k} = 1$ or 0 according as the operation is $\ominus$ or $\oplus$, $n_{k+1} = n_k - f(q_{24-k}) \times 2^{23-k}$. Also $n_{k+1} = n_k \mp x\, 2^{23-k}$ according as $n_k < 0$ or $n_k \geq 0$. Thus $|n_{k+1}| = |x\, 2^{23-k} - |n_k||$.

18/27

and since $x\,2^{23-k} - |n_k| \leq x\,2^{23-k}$ and

$|n_k| - x\,2^{23-k} \leq x\,2^{24-k} - x\,2^{23-k} = x\,2^{23-k}$, we have

$|n_{k+1}| \leq x\,2^{23-k} < 2^{47-(k+1)} - 1$. By this inequality,

$n^{k+1}_{47} = n^{k+1}_{46} = \cdots = n^{k+1}_{46-k}$. Since the next operation is $\ominus$ or $\oplus$ according as $n^{k+1}_{46-k} = 0$ or 1, it is $\ominus$ or $\oplus$ according as $n_{k+1} \geq 0$ or $n_k < 0$.

The algorithm proceeds in this way until finally it will be the case that $|n_{23}| \leq x\,2 < 2^{24} - 1$, $n^{23}_{47} = n^{23}_{46} = \cdots = n^{23}_{24}$ and the operation of substep 23 is $\ominus$ or $\oplus$ according as $n_{23} \geq 0$ or $n_{23} < 0$.

Substep 23: Compute

$$
\begin{array}{lccccccccl}
 & n^{23}_{47} & \cdots\cdots\cdots & n^{23}_{24} & n^{23}_{23} & n^{23}_{22} & \cdots\cdots & n^{23}_{0} & & \leftrightarrow n_{23} \\
\ominus \text{ or } \oplus & 0 & \cdots\cdots\cdots & 0 & x_{23} & \cdots & \cdots\cdots & x_0 & & \leftrightarrow x\,2^0 \\
\hline
= & n^{24}_{47} & \cdots\cdots\cdots & n^{24}_{24} & n^{24}_{23} & n^{24}_{22} & \cdots\cdots & n^{24}_{0} & = N_{24} & \leftrightarrow n_{23} \pm x2^0 = n_{24}.
\end{array}
$$

Since $q_1 = 1$ or 0 according as the operation is $\ominus$ or $\oplus$, $n_{24} = n_{23} - f(q_1)\,x\,2^0$. Also $n_{24} = n_{23} \pm x\,2^0$ according as $n_{23} < 0$ or $n_{23} \geq 0$. Thus $|n_{24}| = |x\,2^0 - |n_{23}||$ and since $x\,2^0 - |n_{23}| \leq x$ and $|n_{23}| - x\,2^0 \leq x\,2 - x\,2^0 = x$, we have $|n_{24}| \leq x \leq 2^{23} - 1$. By this last inequality $n^{24}_{47} = n^{24}_{46} = \cdots = n^{24}_{24} = n^{24}_{23}$.

Recapitulating our results through step 2°, we have

$|n_{24}| \leq x$ and

$N_1 \longleftrightarrow n_1 = a \quad - f(q_0) \times 2^{23}$

$N_2 \longleftrightarrow n_2 = n_1 \quad - f(q_{23}) \times 2^{22}$

$N_3 \longleftrightarrow n_3 = n_2 \quad - f(q_{22}) \times 2^{21}$

.
.
.

$N_k \longleftrightarrow n_k = n_{k-1} - f(q_{25-k}) \times 2^{24-k}$

.
.
.

$N_{24} \leftrightarrow n_{24} = n_{23} \quad - f(q_1) \times 2^{0}.$

Adding the images $n_1 + \dots + n_{23} + n_{24} =$

$$n_1 + \dots + n_{23} + a - f(q_0) \times 2^{23} - \sum_{k=22}^{0} f(q_{k+1}) \times 2^k, \text{ or}$$

$$n_{24} = a - x\left(f(q_0)\, 2^{23} - \sum_{k=22}^{0} f(q_{k+1})\, 2^k \right) \text{ and by}$$

Lemma V and the fact that $q_0 = 1$, $n_{24} = a - x \sum_{k=23}^{0} q_k\, 2^k$.

We know that $|n_{24}| \leq x$. If $n_{24}^{24} = n_{47}^{24} = 0$ then $0 \leq n_{24} < x$. For assume $n_{24} = x$. Then, since the number added to or subtracted from n_{23} in substep 23 is x, the operation must be $\ominus$, for if it were $\oplus$, n_{23} would equal zero and zero calls for subtraction. Now since the operation is $\ominus$, we must have $n_{23} = 2x = x\, 2^1$. In substep 22 the operation must again be $\ominus$, for if it were $\oplus$, we would have, in view of the fact that $x2^1$ is added to or subtracted from n_{22} in substep 22, $n_{22} = 0$ which would call for subtraction. Again since the operation is $\ominus$, we must have $n_{22} = 2(x2^1) = x2^2$. Proceeding in this way, we find that the operation of substep 1 is $\ominus$. But we already know that this operation is $\oplus$ and hence we have a contradiction. Therefore $0 \leq n_{24} < x$. Now if $n_{24}^{24} = 0$ then no action is taken in step 3 and AC contains $N_{24} \longleftrightarrow n_{24}$, and QR contains $Q = q_{23} \cdots q_0 \longleftrightarrow q = q_{23} \cdots q_0$ since $q_{23} = 0$. But $n_{24} = a - x \sum_{k=23}^{0} q_k \, 2^k = a - qx$ or $a = qx + n_{24}$ where $0 \leq n_{24} < x$.

On the other hand if $n_{24}^{24} = n_{47}^{24} = 1$ then $-x \leq n_{24} < 0$. We do know that $a = qx + n_{24}$ where q is the image of the content of QR after step 2°. In step 3°, in this case, $q_0 = 1$ is replaced by $q_{23} = 0$ and QR represents $q' = q - 1$ at the completion of the entire operation. Also MR is added to AC and at the end of the operation AC represents $n_{24} + x$. But $a = qx + n_{24} = (q - 1)\,x + (n_{24} + x)$ where $0 \leq n_{24} + x < x$.

Thus we have proved that if $0 < x \leq 2^{23} - 1$ and $0 < a < x\,2^{23}$ then the division algorithm gives the correct

L-141

result, that the contents of QR, AC and XR correspond respectively to the quotient, the least non-negative remainder and the divisor.

Suppose $a \geq x\,2^{23} > x\,2^{23} - 1 = x(2^{23} - 1) + (x - 1)$. Then the quotient must be greater than $2^{23} - 1$, that is to say, QR is not capable of holding the quotient. If $a \geq x\,2^{23}$ the computer will not give the correct answer, nor will it give an answer correct modulo $2^{24} - 1$.

It remains to extend this result to other cases. In what follows the algorithm will be used as it stands on page 14. The device made possible by Lemmas II and III will not be employed.

Case II: Assume that

$$W \begin{cases} 0 > x \geq 1 - 2^{23} \\ 0 < a < |x|\,2^{23} \\ X = 1x_{22} \ldots x_0 \longleftrightarrow x = -0x'_{22} \ldots x'_0 \\ A = 0a_{46} \ldots a_0 \text{ --- } a = 0a_{46} \ldots a_0 \end{cases}$$

$$\text{then } V \begin{cases} 0 < |x| \leq 2^{23} - 1 \\ 0 < a < |x|\,2^{23} \\ X' = 0x'_{22} \ldots x'_0 \longleftrightarrow |x| = 0x'_{22} \ldots x'_0 \\ A = 0a_{46} \ldots a_0 \longleftrightarrow a = 0a_{46} \ldots a_0 \end{cases}$$

We know from case I that the computer will properly divide a by $|x|$. Call this division V. Call the division of a by x division W. We shall compare division W with division V.

L-141

In each division $a_{47} = 0$ so no action is taken in step 1 of the algorithm. In substep 1 of step 3 in division V, $a_{47} = x'_{23} = 0$ so XR is subtracted into AC and 1 inserted in q_0. In the corresponding step of division W, however, $a_{47} = 0 \neq 1 = x_{23}$ and XR is added into AC and q_0 left as zero. But in W the content of XR is the complement of the content of XR in V, and since addition of XR and subtraction of its complement are the same thing, the content of AC at the end of substep 1 is the same in W as in V. Similarly in each subsequent substep the contents of AC in V and W are the same and opposite digits are inserted in QR.

In step 4 action (a) is taken in both V and W or action (c) is taken in both divisions. If it is (c), no action is taken and the remainder in AC is the same in V as in W and the contents of QR in V and W are complements. If the action is (a), X' is added to AC in each division and in V q_0 is changed from 1 to 0 and in W q_0 is changed from 0 to 1. Thus in either case (a) or (c) the content of AC is the same in V and W, let its image be r. The contents of QR in V and W are complements. If q is the image of the content of QR in V and q' is the image of the content of QR in W, then $0 \leq q \leq 2^{23} - 1$ and $1 - 2^{23} \leq q' \leq 0$ and $q' = -q$.

From case I, $a = q\,|x| + r = (-q)(-|x|) + r = q'\,x + r$.

The algorithm is thus valid in this case as well as case I.

Case III: Let

$$V \begin{cases} 0 < x \leq 2^{23} - 1 \\ 0 > a > x(1 - 2^{23}) \\ X = 0x_{22} \ldots x_0 \longleftrightarrow x = 0x_{22} \ldots x_0 \\ A = 1a_{46} \ldots a_0 \longleftrightarrow a = -0a'_{46} \ldots a'_0 \end{cases}$$

The division of a by x will be referred to as division V. In step 1, since $a_{47} = 1$, XR is subtracted from AC. The resulting digital number is $Z = 1z_{46} \ldots z_0 \longleftrightarrow -0z'_{46} \ldots z'_0 = z = a - x < 0$. Let division W be the division of $|z|$ by x. Now

$$W \begin{cases} 0 < x \leq 2^{23} - 1 \\ 0 < |z| < x2^{23} \quad (\text{Since } |z| = |a-x| \leq |a| + |x| < x(2^{23}-1)+x = x2^{23}.) \\ X = 0x_{22} \ldots x_0 \longleftrightarrow x = 0x_{22} \ldots x_0 \\ Z' = 0z'_{46} \ldots z'_0 \longleftrightarrow |z| = 0z'_{46} \ldots z'_0. \end{cases}$$

In step 1 of division W nothing is done. We will compare steps 3 of divisions V and W. In substep 1 of W XR is subtracted into AC and 1 inserted in q_0. In substep 1 of V, since $z_{47} = 1 \neq 0 = x_{23}$, XR is added into AC and 0 inserted in q_0. Since the contents of AC just before substep 1 in V and W are complementary, and the operations are opposite, the resulting digital numbers are complementary. Thus in substep 2

L-141

the operations are again opposite and opposite integers inserted in QR. The process obviously continues in this manner, and at the end of step 3 the contents of AC in the divisions V and W are complementary and so are the contents of QR.

Let d, -d, q, -q be the images at the end of step 3 of the contents of AC in division W, AC in V, QR in W, and QR in V respectively. In the course of the proof of case I we showed that $|z| = qx + d$ even before step 4 is taken. Therefore $z = -|z| = -qx - d$. Suppose now that in step 4 of division V course (b) is taken. Then AC becomes the image of $-d + x$ and QR remains the image of $-q$. But since $z = a - x$, $a = z + x = (-qx - d) + x = (-q)(x) + (-d + x)$. Of course $0 \leq x - d < x$. Assume on the other hand that course (d) is taken. Then AC remains the image of $-d$ (which in this alternative must be positive) and QR becomes $-q + 1$ since q_0 is changed from 0 to 1 and $-q < 0$. Here $a = z + x = (-qx - d) + x = x(-q + 1) + (-d)$. Thus in either course, at the end of the operation XR contains the divisor, QR the quotient and AC the least non-negative remainder.

<u>Case IV</u>: Let

$$V \begin{cases} 0 > x \geq 1 - 2^{23} \\ 0 > a > |x| (1 - 2^{23}) \\ X = 1x_{22} \ldots x_0 \leftrightarrow x = -0x'_{22} \ldots x'_0 \\ A = 1a_{46} \ldots a_0 \leftrightarrow a = -0a'_{46} \ldots a'_0 \end{cases}$$

L-141

$$\text{then } W \begin{cases} 0 < |x| \leq 2^{23} - 1 \\ 0 > a > |x|\,(1 - 2^{23}) \\ X' = 0x'_{22} \cdots x'_0 \longleftrightarrow |x| = 0x'_{22} \cdots x'_0 \\ A = 1a_{46} \cdots a_0 \longleftrightarrow a = -0a'_{46} \cdots a'_0 \end{cases}$$

We will compare division V, an instance of case IV, with division W, an instance of case III. In step 1 the same action is taken in each division and the contents of AC in each are the same at the end of this step. In substep 1 of step 3 in division W, $a_{47} = 1 \neq 0 = x'_{23}$ so XR is added into AC and 0 inserted in q_0. In the corresponding step of division V, $a_{47} = 1 = x_{23}$ so XR is subtracted into AC and 1 inserted in q_0. Since machine addition and machine subtraction of the complement are the same thing, the contents of AC at the end of substep 1 are the same in each division. The process continues in this way, and at the end of step 3 the contents of AC in V and W are the same and the contents of QR are complementary.

Let r be the image of the contents of AC in each division, let q and -q be the images of the contents of QR in V and W respectively. If in step 4 course (b) is taken in W it is taken in V. Should this be the case, QR remains the same and in each division AC becomes the image of $r + |x|$ which is positive and less than $|x|$. But from case III

26/27

L141.

$a = |x|(-q) + (r - |x|) = (-|x|)(q) + (r + |x|) = xq + (r + |x|)$.

If course (d) is taken in W it is taken in V also. Here AC remains the image of r which must in this case be positive and less than $|x|$. In W, QR becomes the image of $-q + 1$ and, in V, QR becomes the image of $q - 1$. By case III $a = |x|(-q + 1) + r = (-|x|)(q - 1) + r = x(q - 1) + r$. Hence in each case XR is the image of the divisor, QR the image of the quotient, and AC the image of the least non-negative remainder at the end of the operation.

V. Cases involving zero: If 0 is divided by a positive number, the quotient turns out to be 0 and the remainder 0. If 0 is divided by a negative number the remainder is 0 and the quotient is all 1's, a result which is not completely nonsensical. If a positive number $\leq 2^{23} - 1$ is divided by 0 the quotient is all 1's and the remainder is the dividend. If a negative number $\geq 1 - 2^{23}$ is divided by 0 the quotient is 0 and the remainder is the dividend (negative). If 0 is divided by 0 the remainder is 0 and the quotient all 1's. These results are easily verified.

We have proved that if $1 - 2^{23} \leq x \leq 2^{23} - 1$, but $x \neq 0$, and if $|x|(1 - 2^{23}) < a < |x| 2^{23}$, but $a \neq 0$ if $x < 0$, then the algorithm gives the correct result.

If QR becomes filled with 1's through a division involving zero, and QR is then added or subtracted into AC, the content of AC is not altered.

P. Billingsley
April 26, 1949

27/27