Project STRETCH
Link Computer Memo 2

Subject: Division in the Link Computer
By: O. Bedrij, W. Wolensky
Date: September 17, 1956

This memo describes a matrix control decimal and binary divide scheme, which may be used in the Link Computer. The system is based on the range of ramainder possible after a specific dividend divisor relationship has been established. The new method indicates a reduction of speed by at least a factor of 3 over more conventional methods.

The division by matrix control decoder is performed in the following manner: A decoder analyzes the first digit of the dividend and the first digit of the divisor. If the divisor digit is equal to, or smaller than the dividend, subtraction is performed. If the divisor digit is larger, the dividend and quotient register shifts one position left, and the dividend is analyzed versus divisor again. A predetermined (table values) operation is specified for possible combinations of divisor digits versus dividend digits. A positive remainder, after a specified subtraction orders the quotient digit to be increased by the decoder multiple. Reanalysis of the dividend versus divisor digit is again performed.

The cycle of analyzing, n-tupling, subtracting, and shifting, proceeds until the dividend is exhausted and division completed.

The appendix of this memo contains a sample step by step procedure of binary division using the matrix decoders shown as tables in Fig. II and III. The system can be modified so that control decoding problem can be reduced, resulting in an increased time to complete execution of a divide operation. The table can be stated in formula fashion as:

$$n\,(Dr \neq 1) \leq Dd \leq 2n\,(Dr \neq 1) - 1$$

where:

n = possible operation to be performed - 1Dr, -2Dr, -4Dr, -8Dr, shift on zero and error condition.

Dr = most significant divisor digit

Dd = most significant dividend digit at time T.

The over and over subtraction of divisor from dividend is another means of obtaining a quotient. In over subtraction return to positive remainder is accomplished by adding back the divisor before shifting to next cycle. This method, although simple in operation is costly in time (see Fig. 1, Item 2*).

By employing a doubler (Fig. 1, Item 3*) for subtraction, the divide time is somewhat improved. A divide operation takes place with subtraction of double divisor, until over subtraction occurs, in which case, add back double divisor, then subtract one divisor from the dividend. If over subtraction occurs, add back the divisor and shift, if no over subtraction, shift to next cycle.

The use of a doubler and quadrupler (Fig. 1, Item 4*) scheme is illustrated in the table also. The quadrupled divisor is subtracted in the same manner as the doubled divisor in the previous case, until over subtraction occurs, where quadrupled divisor is added and the procedure that follows is the same as above.

Below is a table of comparisons for the mentioned systems:

	ITEMS			
	1*	2*	3*	4*
BINARY	NEW MATRIX SYSTEM	REPETITIVE SUBTRACTION	REPETITIVE SUBTRACTION WITH DOUBLER	REPETITIVE SUBTRACTION DOUBLER QUAD.
ANC	1.32	13.5	8.0	8.5
A	20.6 μs	211 μs	125 μs	132.7 μs
B	31.8 μs	324 μs	192 μs	200 μs
DECIMAL				
ANC	1.8	5.5	6.5	7.7
A	28.2 μs	85.8 μs	101.5 μs	120 μs
B	43.3 μs	132 μs	156 μs	184.9 μs

$T = t\,(1 + (1 + Nr)\,(Nd - Nr)$ (Average number of cycles)

where:

ANC = average number of cycles

A = 96 bits ÷ 48 bits (24, 4 bits groups vs. 12, 4 bits groups) providing 12-4 bit quotient.

B = 120 bits ÷ 60 bits (30, 4 bits groups vs. 15, 4 bits groups) providing 15-4 bit quotient.

t = time to complete a simple four bit addition, assumed 0.1 μs.

Nr = number of 4 bit bites in divisor
Nd = number of 4 bit bites in dividend

Note, one extra cycle time should be added to each time calculation to account for preliminary test divide.

The new system although composed of simple operations; such as shift one right or left, minus one, and minus two, times the divisor in the decimal, and minus 1, 2, 4, 8, times the divisor in the binary and plus one divisor, becomes complex in the definition, control, and equipment necessary to provide proper divide execution.

In performing a division, a quotient will be generated by a method of repeated subtraction. The quotient length may be equal to the number of digits difference between the dividend and divisor, or the number of digits difference plus one, depending upon the initial value relationship of the dividend and divisor.

As a general rule, over subtraction is permitted only in special cases (when most significant divisor and dividend digits are the same. The special cases that exist, which permit over subtraction, do so for the explicit purpose of preventing a propagation through more than one quotient digit position. The division by or into zero will automatically be detected. At the completion of a divide operation, indicated by a counter, a full remainder will be available for possible further use.

Rounding of the quotient may reasonably be provided automatically if the rounding will not generate a propagation through more than one quotient digit. (Provision for unrestricted rounding can be made at rather high cost of equipment and time, it is presently suggested that if required, the rounding be programmed.) It is precluded at this point that a separate adder will be provided for binary and decimal operations and will provide a 4 bit addition in 0.1 μs.

To conclude, if speed is the main thing, the new system is best so far. A compromise can be made in new system by decreasing size of matrix decoder at a cost increasing the time of execution

OB:bl

APPENDIX

Binary Division (Step by Step Procedure)

Dd = 345,689,421 Dr = 499

Step 1
a) Step thru to find first digit of Dd "3"
b) Shift to find 1st digit of Dr "4"
c) Analyze Dd = 03, Dr = 4; according to table II perform a shift Dd to left one position, also shift Q_1 to Q_2 position, reanalyze dividend.
d) From chart II Dd = 34, Dr = 4 perform - 4Dr opn.
e) Subtract -4Dr from Dd, resulting Dd +. Add 4 to Q_1, and reanalyze.

Step 2
a) Dd = 14; Dr = 4; the table shows -2 opn.
b) Subtract -2Dr from dividend, resulting Dd +. Add 2 to Q_1, and reanalyze again.

Step 3
a) Dd = 04; Dr = 4; table indicates -1Dr opn.
b) Subtract -1Dr from Dd, resulting Dd is minus.
c) Do not add 1 to Q_1.
d) Shift dividend one position left as well as Q_1 into Q_2 (because Dd is minus Q_1 is considered to be 10)

Step 4
a) Since Dd is minus, add + 1Dr to Dd, at the same time making Q_1 = 9.
b) New Dd is plus, reanalyze Dd vs. Dr.
Dd = 01; Dr = 4, shift Dd and Q's left one position and reanalyze.

Step 5
a) Dd = 13; Dr = 4; table indicates - 2Dr opn.
b) Subtract - 2Dr from Dd, resulting Dd plus, so add 2 to Q_1.
c) Reanalyze Dd = 03, Dr = 4 table indicates shift Dd one position left, also shift Q_1 one position left.

Step 6
a) Dd = 38; Dr = 4; table shows -4Dr opn.
b) Subtract - 4Dr from Dd, resulting Dd plus add 4 to Q_1.
c) Reanalyze

Step 7
a) Dd = 12; Dr = 4, do - 2Dr opn.
b) Subtract -2Dr from Dd, Dd is + , add 2 to Q_1.
c) Reanalyze'

Step 8
a) Dd = 08; Dr = 4; perform - 1Dr opn.
b) Subtract - 1Dr from Dd, Dd is plus, add 1 to Q_1.
c) Reanalyze, Dd = 03; Dr = 4 perform one left shift in Dd and Q's.

Step 9

a) Dd = 32; Dr = 4; do - 4Dr opn.
b) Subtract -4Dr from Dd, Dd is + add 4 to Q_1.
c) Reanalyze

Step 10

a) Dd = 12; Dr = 4; according to table Do - 2Dr opn.
b) Subtract -2Dr from Dd, result of Dd +, add 2 to Q_1.
c) Reanalyze Dd vs. Dr, table indicates one left shift for Dd = 02 and Dr = 4. Also shift Q's one position left.

Step 11

a) Because Dd = 21, and Dr = 4 do - 4Dr opn.
b) Subtract - 4Dr from Dd. Result is plus, add 4 to Q_1.
c) Reanalyze Dd vs. Dr. For Dd = 01 and Dr = 4 get ready to shift, but Dd is exhausted (counter shows no more Dd digits): End of divide operation.
d) Dd is the remainder.
e) Shift Q_1 to join other quotient digits in output memory area for register.
f) Execution completed, after placing proper sign notations of required and not previously done.

OB:bl
9/5/56

BINARY DIVISION

Dr = 499

												Q_2	Q_1
Step 1		Dd	0	3	4	5	6	8	9	4	2	1	0	0
		Dd	3	4	5	6	8	9	4	2	1			
	-4	Dr	1	9	9	6								
			1	4	6	0	8	9	4	2	1		0	4
Step 2		Dd	1	4	6	0	8	9	4	2	1			
	-2	Dr		9	9	8								
			0	4	6	2	8	9	4	2	1		0	6
Step 3		Dd	0	4	6	2	8	9	4	2	1			
	-1	Dr		4	9	9								
			-9	9	6	3	8	9	4	2	1		6	10
Step 4		Dd	-9	9	6	3	8	9	4	2	1			
	÷1	Dr			4	9	9							
			0	0	1	3	7	9	4	2	1		6	9
Step 5		Dd	1	3	7	9	4	2	1				6	9 0
	-2	Dr		9	9	8								
			0	3	8	1	4	2	1				6	9 2
Step 6		Dd	3	8	1	4	2	1					6	9 2 0
	-4	Dr	1	9	9	6								
			1	8	1	8	2	1					6	9 2 4
Step 7		Dd	1	8	1	8	2	1						
	-2	Dr		9	9	8								
			0	8	2	0	2	1					6	9 2 6
Step 8		Dd	0	8	2	0	2	1						
	-1	Dr		4	9	9								
			0	3	2	1	2	1					6	9 2 7
Step 9		Dd	3	2	1	2	1						6	9 2 7 0
	-4	Dr	1	9	9	6								
			1	2	1	6	1						6	9 2 7 4

								Q_2 Q_1
Step 10	Dd	1	2	1	6	1		69276
-2	Dr		9	9	8			
		0	2	1	8	1		
Step 11	Dd	2	1	8	1			692760
-4	Dr	1	9	9	6			
		0	1	8	5			692764

11 Cycle per six digit quotient

11/6 = 1, 83 cycles/quotient digit

01 says shift, however, countings have expired in length of field:, problem ended and 185 is the remainder.

DECIMAL DECODING MATRIX

Table I

Divisor Digit

Dividend Digit	0	1	2	3	4	5	6	7	8	9
00		0								
01		- 1	0			Shift one Left				
02		- 1	- 1	0						
03		- 1	- 1	- 1	0					
04		- 2	- 1	- 1	- 1	0				
05		- 2	- 1	- 1	- 1	- 1	0			
06		- 2	- 2	- 1	- 1	- 1	- 1	0		
07		- 2	- 2	- 1	- 1	- 1	- 1	- 1	0	
08		- 2	- 2	- 2	- 1	- 1	- 1	- 1	- 1	0
09		- 2	- 2	- 2	- 1	- 1	- 1	- 1	- 1	- 1
10			- 2	- 2	- 2	- 1	- 1	- 1	- 1	- 1
11			- 2	- 2	- 2	- 1	- 1	- 1	- 1	- 1
12			- 2	- 2	- 2	- 2	- 1	- 1	- 1	- 1
13			- 2	- 2	- 2	- 2	- 1	- 1	- 1	- 1
14			- 2	- 2	- 2	- 2	- 2	- 1	- 1	- 1
15			- 2	- 2	- 2	- 2	- 2	- 1	- 1	- 1
16			- 2	- 2	- 2	- 2	- 2	- 2	- 1	- 1
17			- 10	- 2	- 2	- 2	- 2	- 2	- 1	- 1
18			- 10	- 2	- 2	- 2	- 2	- 2	- 2	- 1
19			- 10	- 2	- 2	- 2	- 2	- 2	- 2	- 1
2x				- 2	- 2	- 2	- 2	- 2	- 2	- 2
3x					- 10	- 2	- 2	- 2	- 2	- 2
4x						- 10	- 2	- 2	- 2	- 2
5x							- 10	- 10	- 2	- 2
6x								- 10	- 10	- 2
7x									- 10	- 10
8x										- 10
9x										

9/12/56 OB:bl

Divisor Digit

Dividend Digit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
00		0														
01		-1	0													
02		-1	-1	0												
03		-1	-1	-1	0					Shift Dd and Q's						
04		-2	-1	-1	1	0				one left						
05		-2	-1	-1	-1	-1	0									
06		-2	-2	-1	-1	-1	-1	0								
07		-2	-2	-1	-1	-1	-1	-1	0							
08		-4	-2	-2	-1	-1	-1	-1	-1	0						
09	4	-4	-2	-2	-1	-1	-1	-1	-1	-1	0					
10		-4	-2	-2	-2	-1	-1	-1	-1	-1	-1	0				
11		-4	-2	-2	-2	-1	-1	-1	-1	-1	-1	-1	0			
12		-4	-4	-2	-2	-2	-1	-1	-1	-1	-1	-1	-1	0		
13		-4	-4	-2	-2	-2	-1	-1	-1	-1	-1	-1	-1	+1	0	
14		-4	-4	-2	-2	-2	-2	-1	-1	-1	-1	-1	-1	+1	-1	0
15		-4	-4	-2	-2	-2	-2	-1	-1	-1	-1	-1	-1	+1	-1	-1
16	8	-8	-4	-4	-2	-2	-2	-2	-1	-1	-1	-1	-1	+1	-1	-1
17		-8	-4	-4	-2	-2	-2	-2	-1	-1	-1	-1	-1	+1	-1	-1
18		-8	-4	-4	-2	-2	-2	-2	-2	-1	-1	-1	-1	+1	-1	-1
19		-8	-4	-4	-2	-2	-2	-2	-2	-1	-1	-1	-1	+1	-1	-1
20		-8	-4	-4	-4	-2	-2	-2	-2	-2	-1	-1	-1	+1	-1	-1
21		-8	-4	-4	-4	-2	-2	-2	-2	-2	-1	-1	-1	+1	-1	-1
22		-8	-4	-4	-4	-2	-2	-2	-2	-2	-2	-1	-1	+1	-1	-1
23		-8	-4	-4	-4	-2	-2	-2	-2	-2	-2	-1	-1	+1	-1	-1
24		-8	-8	-4	-4	-4	-2	-2	-2	-2	-2	-2	-1	+1	-1	-1
25		-8	-8	-4	-4	-4	-2	-2	-2	-2	-2	-2	-1	+1	-1	-1
26		-8	-8	-4	-4	-4	-2	-2	-2	-2	-2	-2	-2	+1	-1	-1
27		-8	-8	-4	-4	-4	-2	-2	-2	-2	-2	-2	-2	+1	-1	-1
28		-8	-8	-4	-4	-4	-4	-2	-2	-2	-2	-2	-2	+2	-1	-1
29		-8	-8	-4	-4	-4	-4	-2	-2	-2	-2	-2	-2	+2	-1	-1
30		-8	-8	-4	-4	-4	-4	-2	-2	-2	-2	-2	-2	+2	-2	-1
31		-8	-8	-4	-4	-4	-4	-2	-2	-2	-2	-2	-2	+2	-2	-1

9/6/56 08:51