

8 Neighbor Sum, Full and indexing scheme. (Chain indexing)

x+3 Load acc with zero

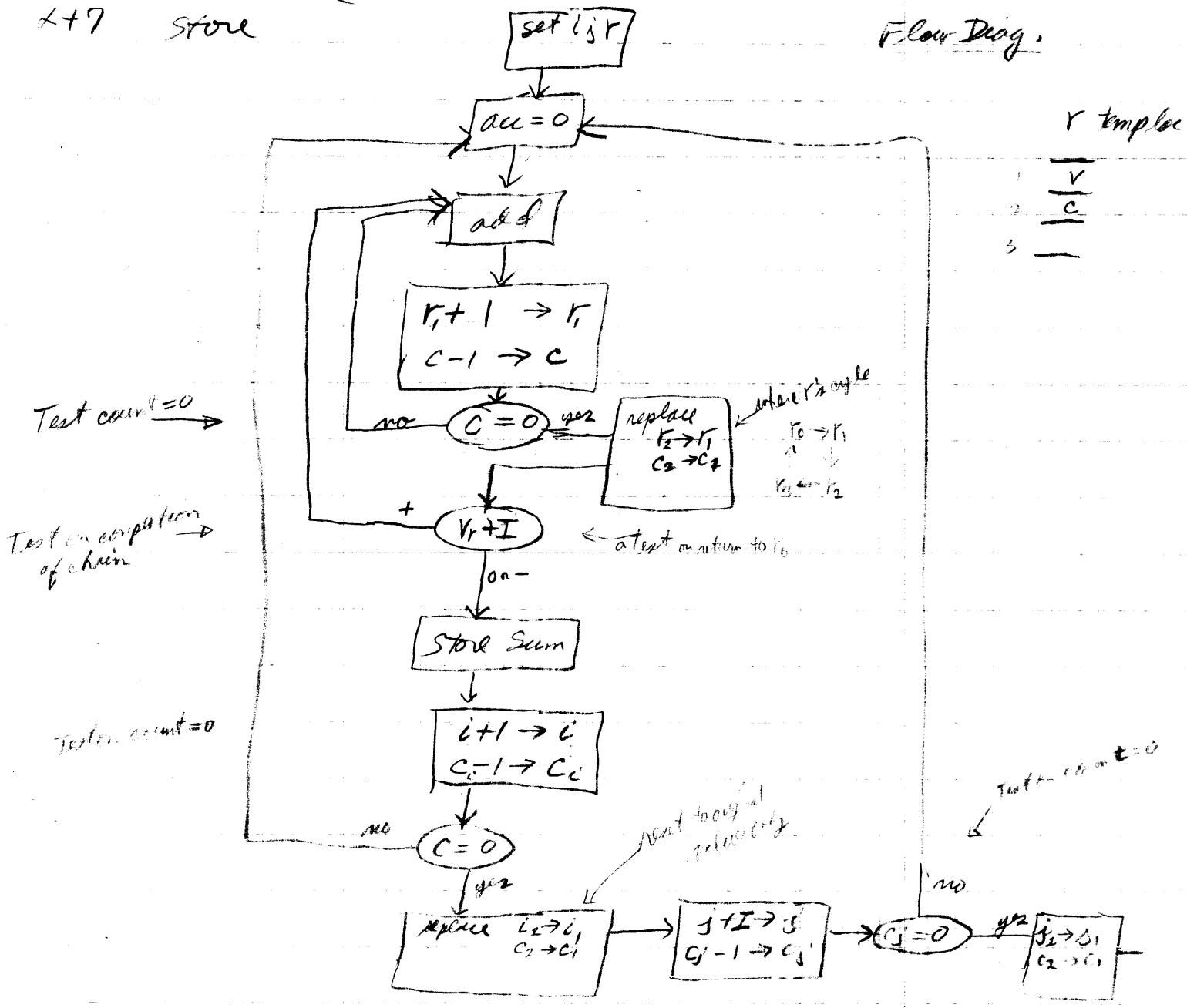
x+4 Add $I(x_{00}) + C(i) + C(j) + C(k)$

x+5 Inc B1. & Dec C $\left\{ \begin{array}{l} V + \Delta \xrightarrow{iSA} V \quad \text{Increment index } r \\ C - 1 \rightarrow C \quad \text{count down } C \end{array} \right.$
 Test if $C=0$ proceed and replace index r by index given by R
 if $C \neq 0$ Branch to x+4

x+6 BVGN $\left\{ \begin{array}{l} \text{Test if } V_r > -\Delta \xrightarrow{iSA} \text{ Branch to } x+4 \\ V_r \leq -\Delta \text{ proceed} \end{array} \right.$

x+7 store

Flow Diag.



Full Note: 8 Neighbor Sum using Reset Addr, and Geometric Addressing as in full md case. [Oct 21, 57]

	Op	I ₁	I ₂	Addr.	
1	Load II		0	$L(J-2)_m$	count for j
2	Load II		j	$L(I)_m$	orig value j
3	Load II		n	$L(I-2)_m$	count for i
4	Load II		i	1	orig value i
5	Load II		*+3	R ₀ , m	orig location for r
6	Load I geo	m		i, s	$C(i) + C(i) \rightarrow C(m)$
7	Load II		l	3	3 → l count for r
11	Fl Load			0	
12	Load I geo	k		m, k	$C(i) + C(j) + C(k) \rightarrow C(l)$
13	Fl Add	k		$L(X_{00})$	
14	Incr.	k		1	k+1 → k
15	Count + Br. No.		r+2		c-1 → c
8	Load Name II	*+3	k	0	$\begin{cases} V_2 \rightarrow C(k) \\ R_1 \rightarrow C(k+1) \\ C_2 \rightarrow C(k+2) \end{cases}$
9	Load II	*+1	*+2	1	
10	Load II	*+1	*+3	2	$R_2 \rightarrow C(k+3)$
16	Count + Br. No.				l-1 → l
17	Fl Store	m		$L()$	Store sum
18	Incr I	i		1	i+1 → i
19	Count + Br. No.	n			n-1 → n
20	Incr I	j		$L(I)$	j+I → j
21	Count + Br. No.	0			

$r = \text{count } 3, 2, 1, 0$
 $i =$
 $j =$
 $l = \text{count } 3, 2, 1, 0$
 m
 Working locations
 $r \alpha: V_1$
 $*+1: R_0 = F(V_1)$
 $*+2: C_1$
 $*+3: R_1 = F(V_2)$
 $R_2, R_3, \text{Temp.}$
 $I_1: V_1$
 $I_1+1: C_1$
 $I_1+2: R_1 (= I_2)$
 $I_2: V_2$
 $I_2+1: C_2$
 $I_2+2: R_2$

V	0123	i, r
C	0123	s, r+1
R	0123	l, r+2
		r+3
		m
		m
		0

Note: The extra count indices could be omitted if an extra count were not given each time.

Note: Reset closing of loop on r is not necessary here since r=0 has to test to store sum anyway, but is included in to make the comparison complete.

XN

pa.

Sept

Example of
3h in index
a half word
scheme

8 neighbor sum

sept 27

→ Transmit 0		acc.	
→ add	Y	I(X00)	$C(y) = i + j + r$ $C(x) = \text{counter}$
incr	Y	1	
incr	X	1	
XNR Compare		1	$(\text{prog} = C(x) \text{ and } C(R+1))$
BR (none)		()	
XNR Reset		I	puts in XNR the $C(R+3)$
X Reset		0	puts in X $C(R)$
Compare	X	I	
BR (none)		()	
STORE	Y		$C(y) = C + 3$

loc	contents	
R ₀	-(I+1)	V
R ₀ +1	3	C
R ₀ +2	R ₁	R
R ₁	-1	V
R ₁ +1	1	C
R ₁ +2	R ₂	R
R ₂	+1	V
R ₂ +1	1	C
R ₂ +2	R ₃	R
R ₃	I-1	V
R ₃ +1	3	C
R ₃ +2	R ₀	R

J ₀	+I	V
J ₀ +1	J-2	C
I ₀	1	V
I ₀ +1	I-2	C

if reset X

not ch.

Half Word with Full Wd. Indexing

OP	IL	Imm	addr
Load 7		i	$f(i)$
Store		i	$f(i)$
Load 8		r	$f(R_i)$
Load I go. m			i, s
LD, Store			0
→ Load I go. R	R		m, r
→ Store	R		$f(R_{00})$
Incr	R		
Br, No	R		
Br, Y	R		
Comp	R		$f(i)$
{ Comp	R		$f(i)$
{ Br, No			
LD Store	m		$f(i)$
Store	i		/
Comp			
Branch			
Comp			
Branch			

18. where (Comp can be replaced by Count & Br, at 20. Super 1.5.10)

(store loads: Same No. of ops but faster)

V, C

Fl Load

0

Fl add R
Add Imm, L R
Count + Br No R
Count Br No + R L
Fl Store m
Add Imm J
Count + Br No J
Add Imm .

L(X00)
1
(13)
(8)
L(X00)
I
(6)

Fl Load

0

→ Fl Add R L(X00)
Incr + C + R R
Br Non 0