Lecture Notes   Feb 16, 1959

Asynchronism and Overlapping :

How does one make a faster computer ?

(1) Faster components
  (a) logical elements
  (b) memory elements

(2) Improved instructions (example :

(3) Machine organization

  704 (review charts)

$-C(A+B)$ / D

CLA	A	CLA	A
ADD	B	ADD	B
(STO / [illegible])			
[illegible]	C	MPY	C
(CHS)		DIVI	D
(STO)	T		
(CLA)	D		
DIV	T		

→ Review STRAP mnemonics

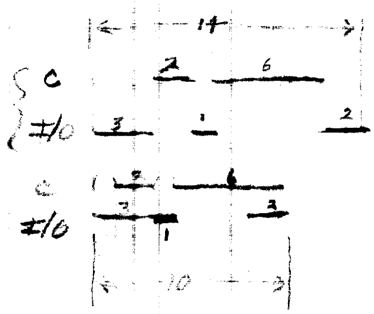
later [Draw charts of
  7090
  LARC
  STRETCH

Point out Type of overlap.

(1) I/O – Compute overlap.

(note conversion is overlapped in 704)

penalty : logic of code
  hardware for channels



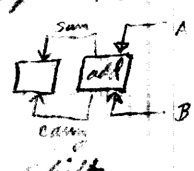
28% [illegible]

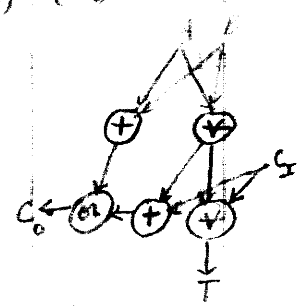
Useful calc = 8
IO burden = 6 → 2
burden (+75%) (25%)

(2) Decrease steps for individual operation

Example : adder serial adder

n. Δt





penalty

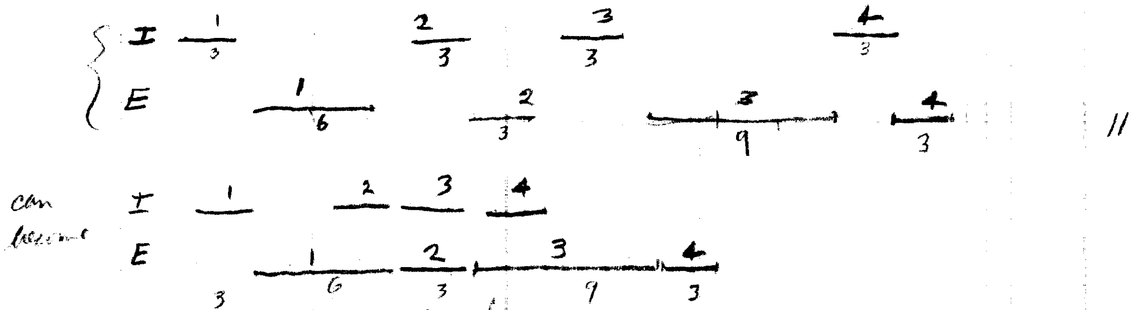
- make more parallel Δt + b (carry prop burden)
- make skip 0's & 1's
- make carry stopping.

704 { Fixed Add 2 24 μs
  { Fl Add 7 84 μs

– mpy – repeated addition
serial – serial
serial – parallel
parallel – serial
parallel – parallel
carry stop

(3) Indexing – Execution overlap.

penalty – { 2 memories
branching – extra equipment
logic hazards.
time not all overlapped.



useful time 7 7
indexing burden 1 ← 4
(+14%) (+58%)

→ Draw LARC chart
Buffer Registers

(4) Instruction counter separate

(5) Fetching 2 instructions at a time
- penalties, gains

(6) Multiple Memories
- gain – higher effective rate
- penalty – possible conflicts
- logical errors possible

(7) Buffering (Look-ahead)

	overlap			
comparison: on particular prob	1	7090' (with faster AO)	~ 10.	
	2	LARC' (LARC with HSTR)	27.	(2.7 faster)
	8	SIGMA	64.	(2.4 faster)