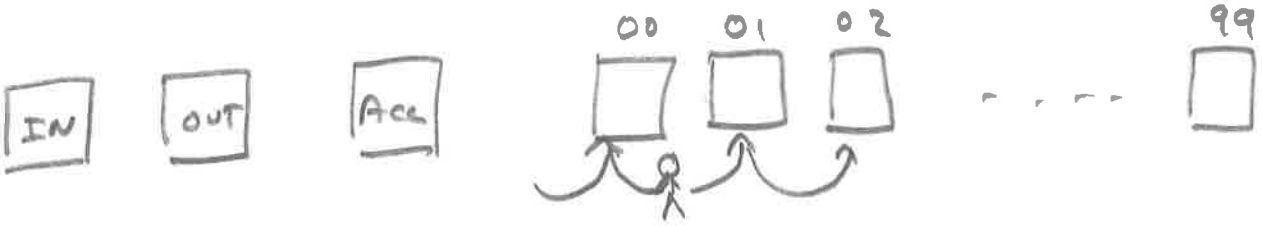
Microprocessors and Assembly

Little Man Computer



Input → user input

Output → e.g. to screen

Acc → accumulator (for addition + subtraction)

Program Counter → starts at zero, increments prior to each instruction, but may be overwritten by the instruction.
(Next instruction taken from address in PC)

Mailboxes: Stores data and program.

LMC Instruction Set

1XX	ADD	Add value in mail box XX to accumulator
2XX	SUB	Subtract value "
3XX	STA	Store contents of Acc into mail box XX.
5XX	LDA	Load contents of XX into the accumulator
6XX	BRA	Set PC to mail box XX
7XX	BRZ	Set PC mail box XX if Acc is zero
8XX	BRP	Set PC mail box XX if Acc is zero or positive
901	INP	load input to Acc
902	OUT	Send Acc to Out
000	COB	Coffee Break

Example: Adder (A+B)

```
901     INP
350     STA  FIRST
901     INP
150     ADD  FIRST
902     OUT
000     COB
FIRST   DAT
```

Example Subtracter (A - B)

```
901     INP
350     STA  FIRST
901     INP
351     STA  SECOND
550     LDA  FIRST
251     SUB  SECOND
902     OUT
000     COB
```

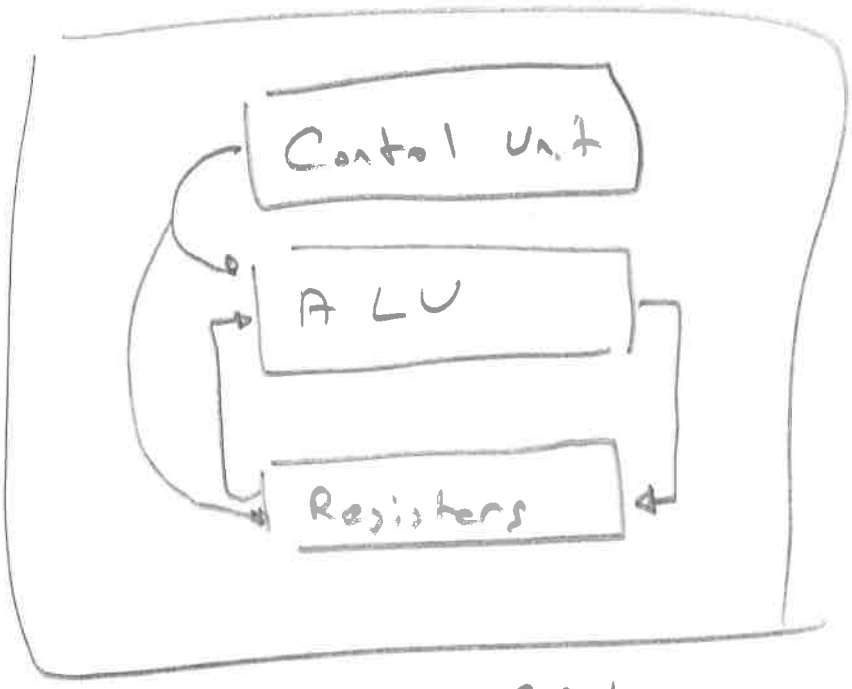
Example multiplier

```
00   901   INP
01   350   STA A
02   901   INP
03   351   STA B
04   712   LOOP BRZ DONE
05   552   LDA SUM
06   150   ADD A
07   352   STA SUM
08   551   LDA B
09   253   SUB ONE
10   351   STA B
11   604   BRA LOOP
12   552   DONE LDA SUM
13   902   OUT
14   000   COB
...
50   -     A DAT
51   -     B DAT
52   0     SUM DAT 0
53   1     ONE DAT 1
```

Microprocessor : single IC with CPU

Central Processing Unit : carries instructions

in computer program



Bus:

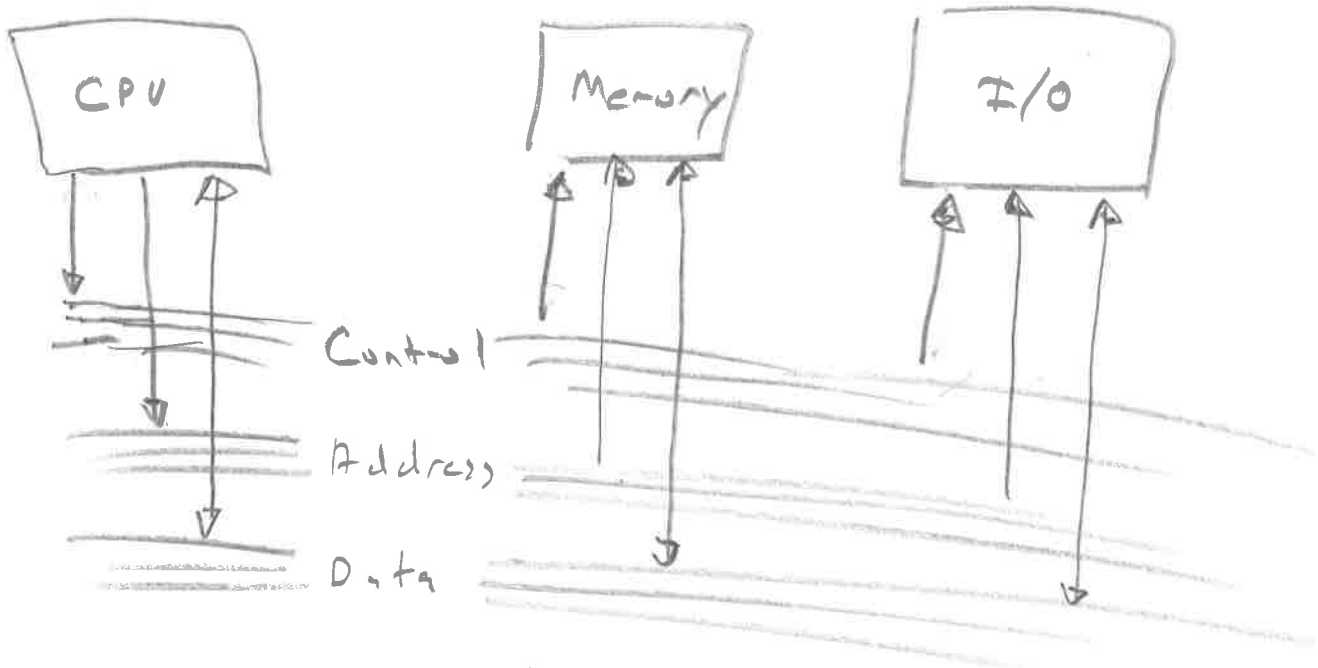
FPGA → each transistor can have single fixed role → grid architecture

Microprocessor:

Shared resources, need way to move data back and forth:

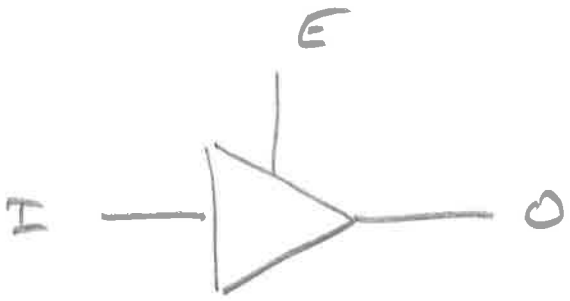
Generic Bus:

In computer :



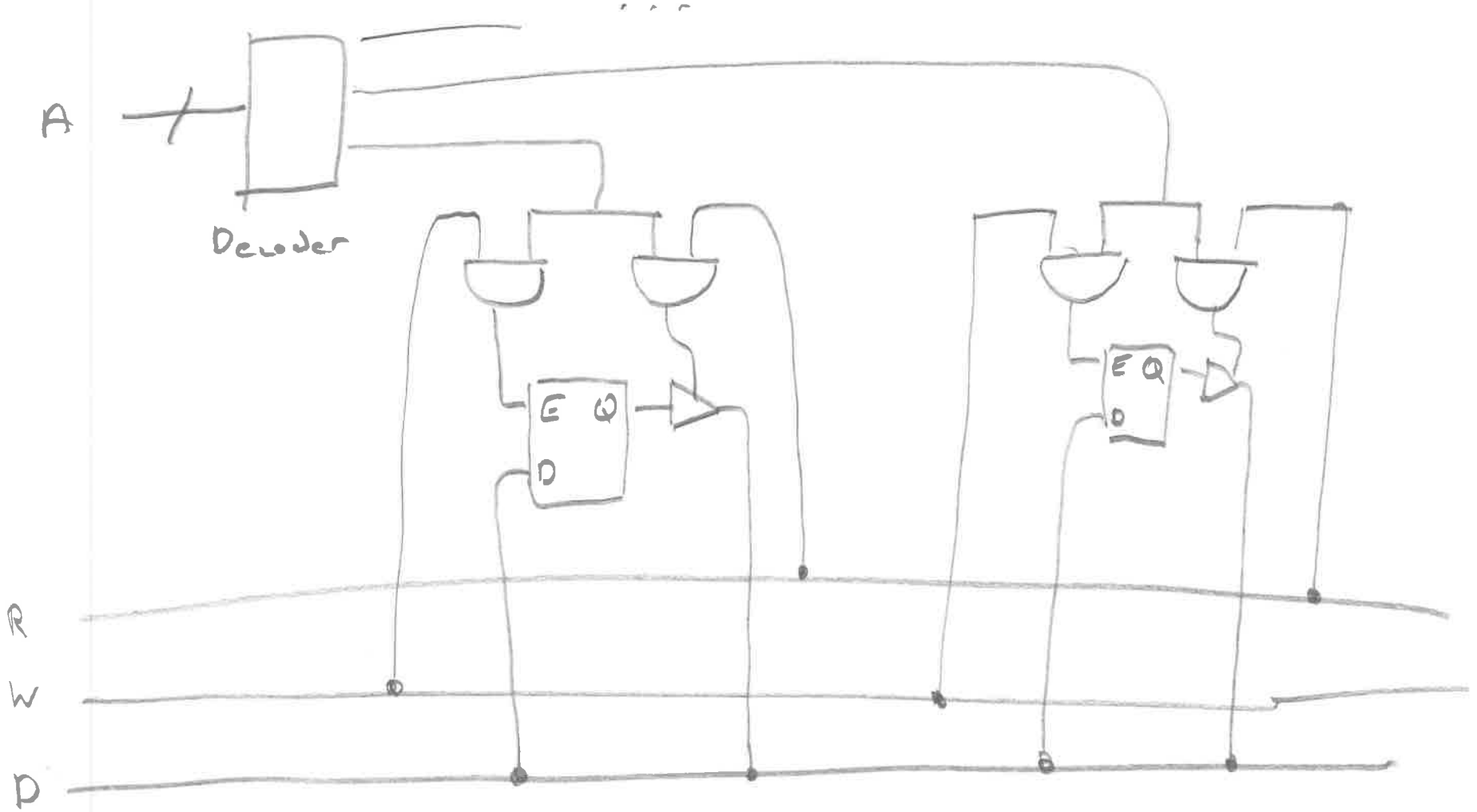
(Simplest Architecture)

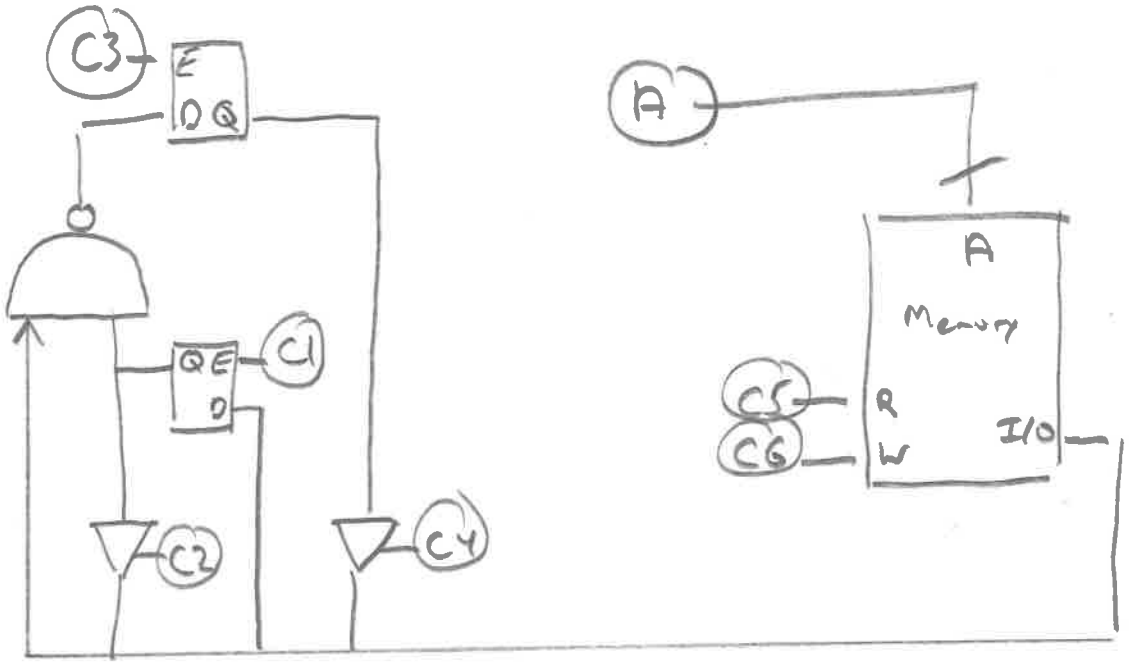
Tri - State Buffer



I	E	O
X	0	Z
Y	1	Y

Memory

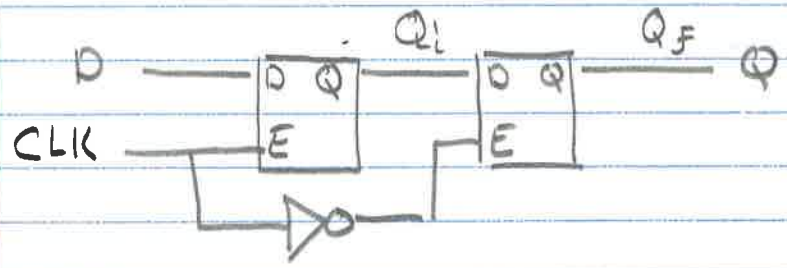




1-bit asynchronous, NAND computer

		A	C1	C2	C3	C4	C5	C6	
		X	/	0	0	0	/	0	
LDA	X	X	0	/	0	0	0	/	
STA	X	X	0	0	/	0	/	0	(1)
WAND	X	—	/	0	0	/	0	0	(2)

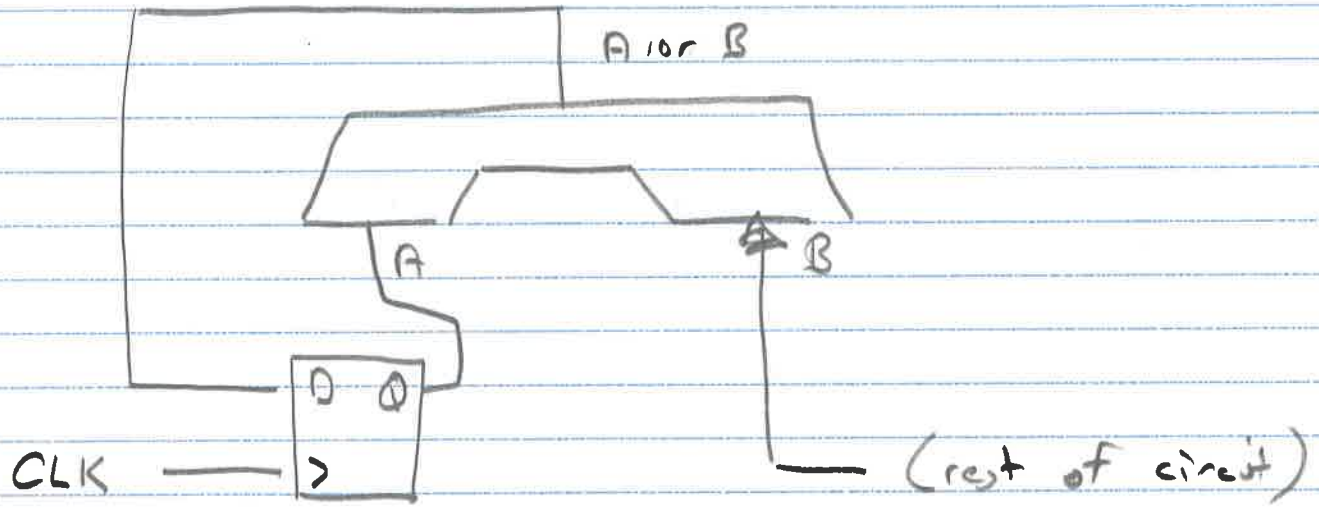
Edge - Triggered D - flip flop

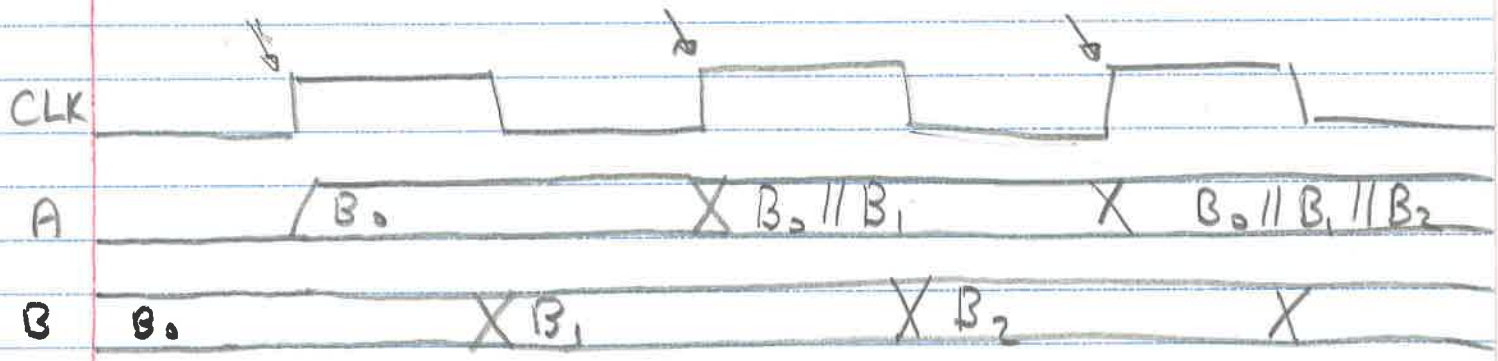


* Won't see new value until after falling edge, ie. Next Clock Cycle

Symbols

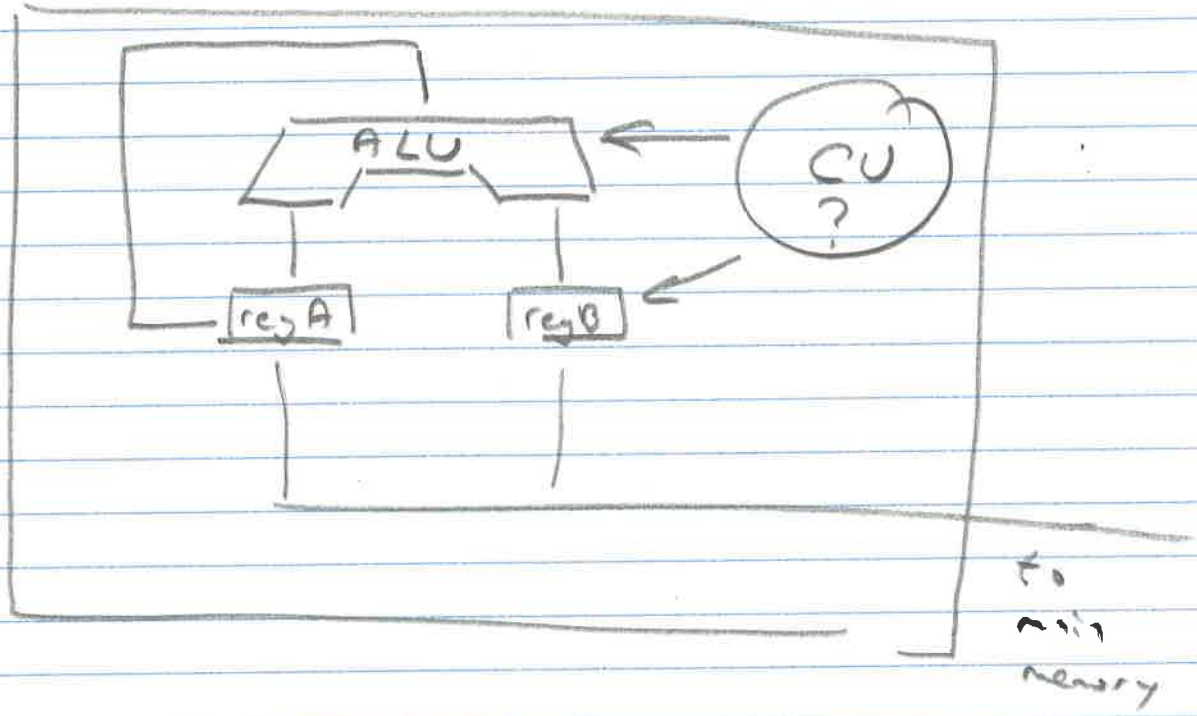
Synchronous Accumulator





As long as we feed B $\frac{1}{2}$ cycle ahead, accumulator handles one new input per cycle.

CPU: Central Processing Unit



By now understand workings of ALU for each assembly command as a fixed set of steps:

→ moving data into registers
→ digital synchronous logic

all driven by control signals (Ci)

Fixed Program Computer

Early computers applied a fixed set of steps to variable input data

C_1	0	0	1	Known by Designer
C_2	0	1	0	
⋮				
C_N	0	0	1	
R1	?	?	?	Computed at run time.
R2	?	?	?	
R3	?	?	?	

First computers had a hard-wired CU, no easy way to change it.

Turing / Von Neumann / Zuse developed idea to

Store Instructions in Memory.

Incredibly powerful:

→ General Purpose Computer

→ New Algorithms Possible based on recursion (GOTO and branching conditional)