

Koodi täitmine protsessoris

Märt Kalmo

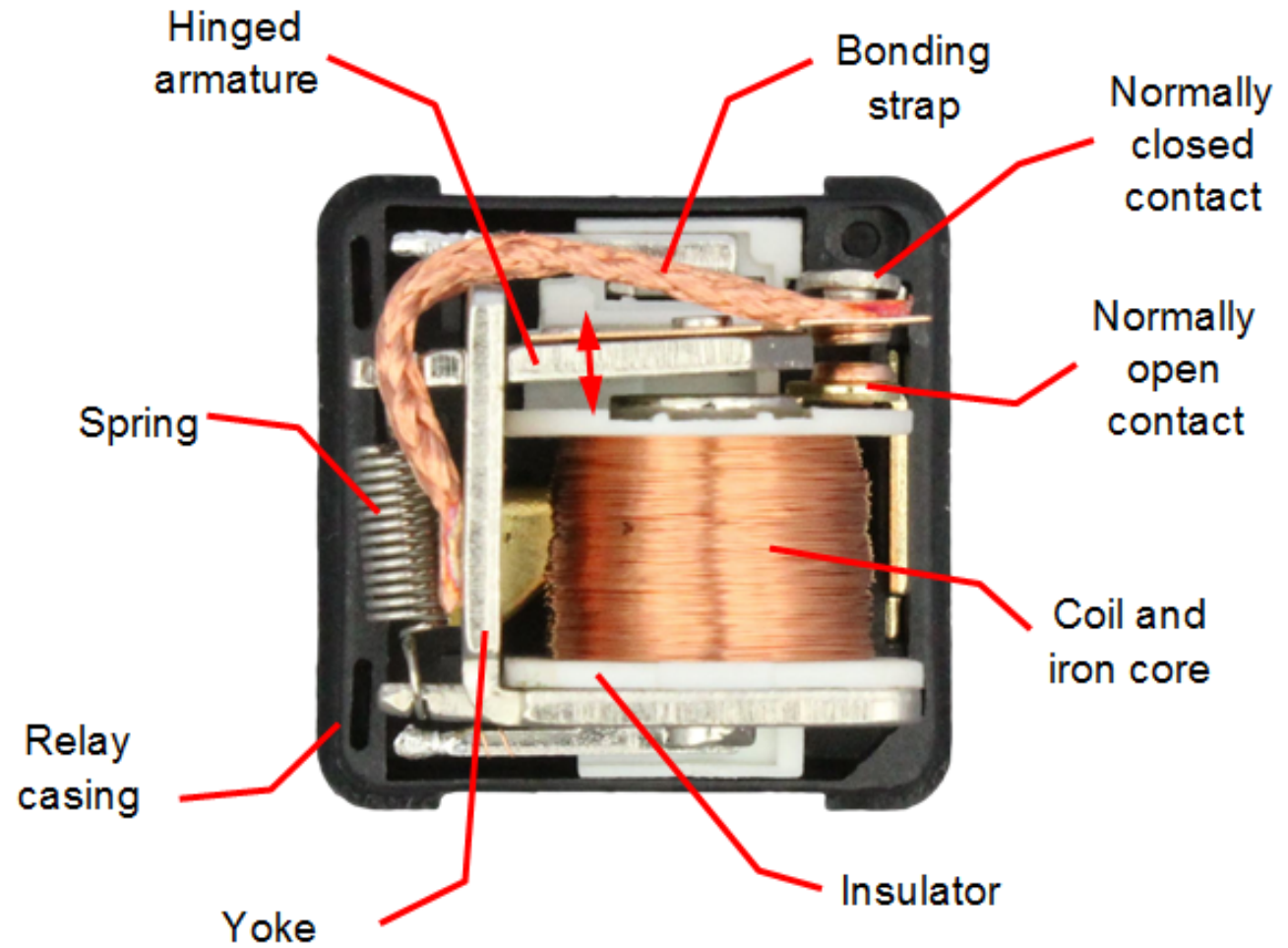
Lülitid ja energia



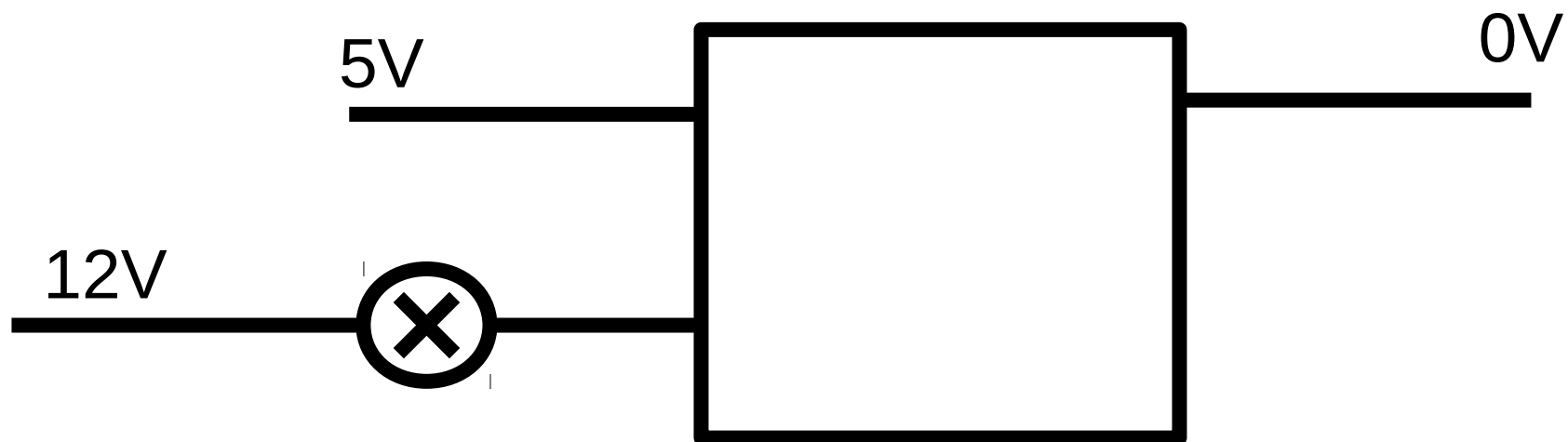
Ajalugu

- 1493 Leonardo da Vinci
- 1666 Gottfried Leibniz (binaarne)
- 1815 George Boole
- 1941 Z3 (Turing-complete, reeledest)

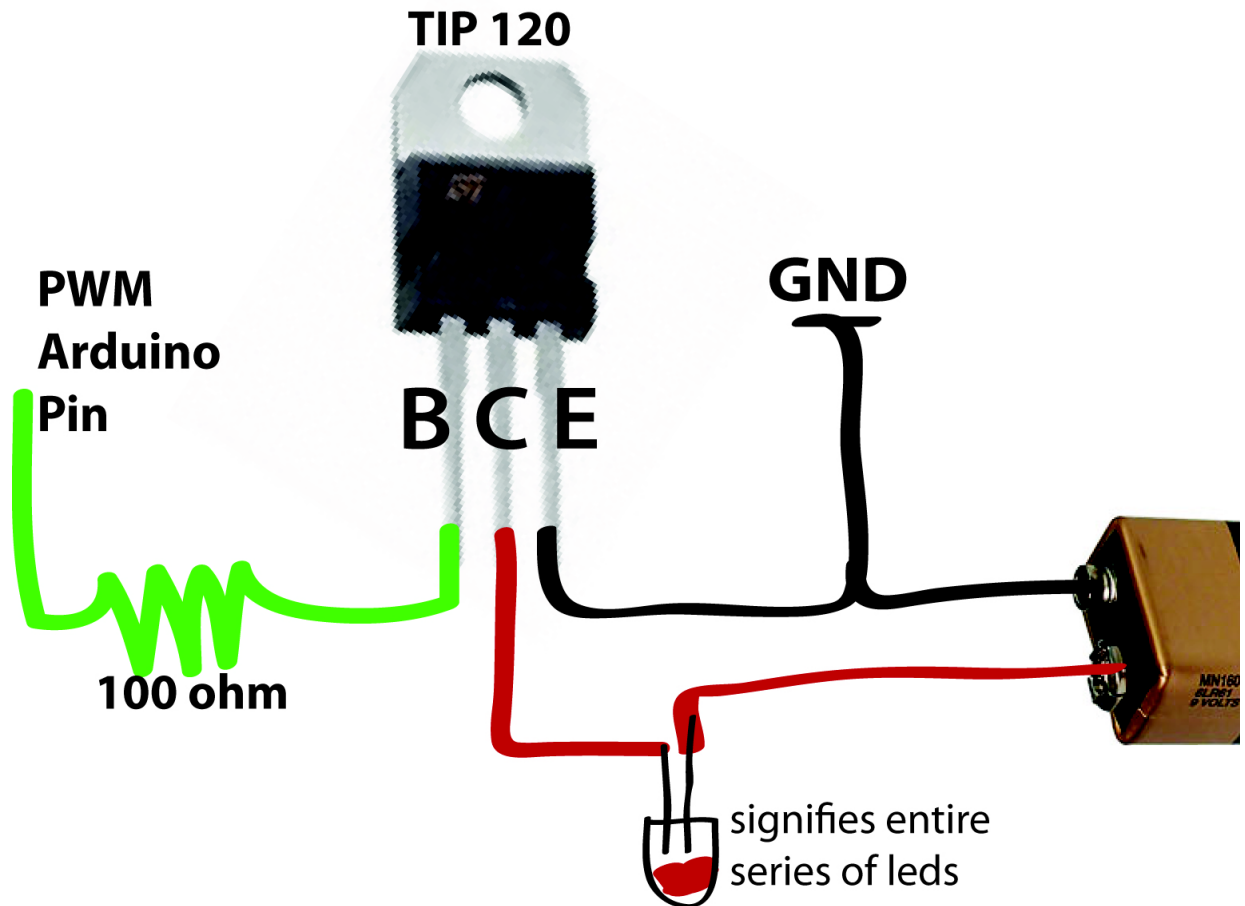
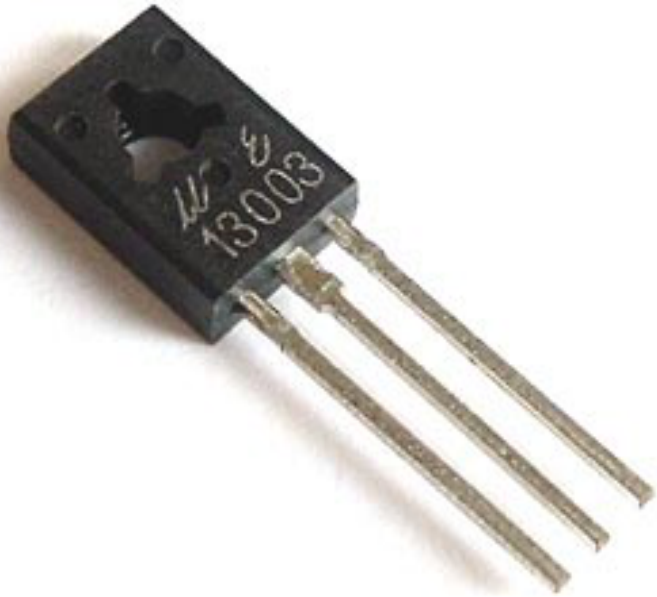
Relee



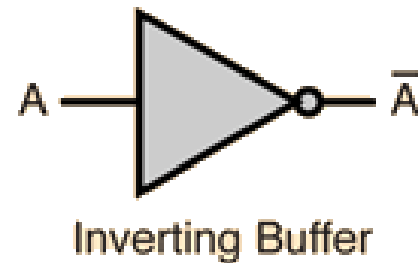
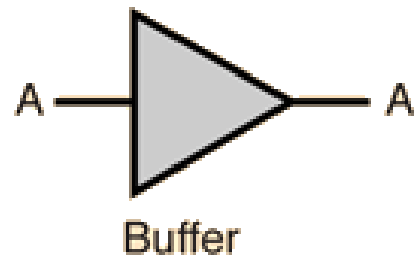
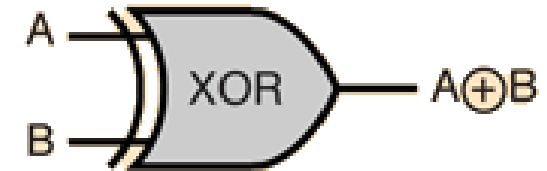
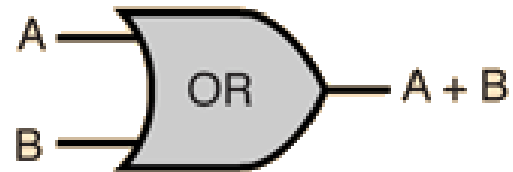
Relee



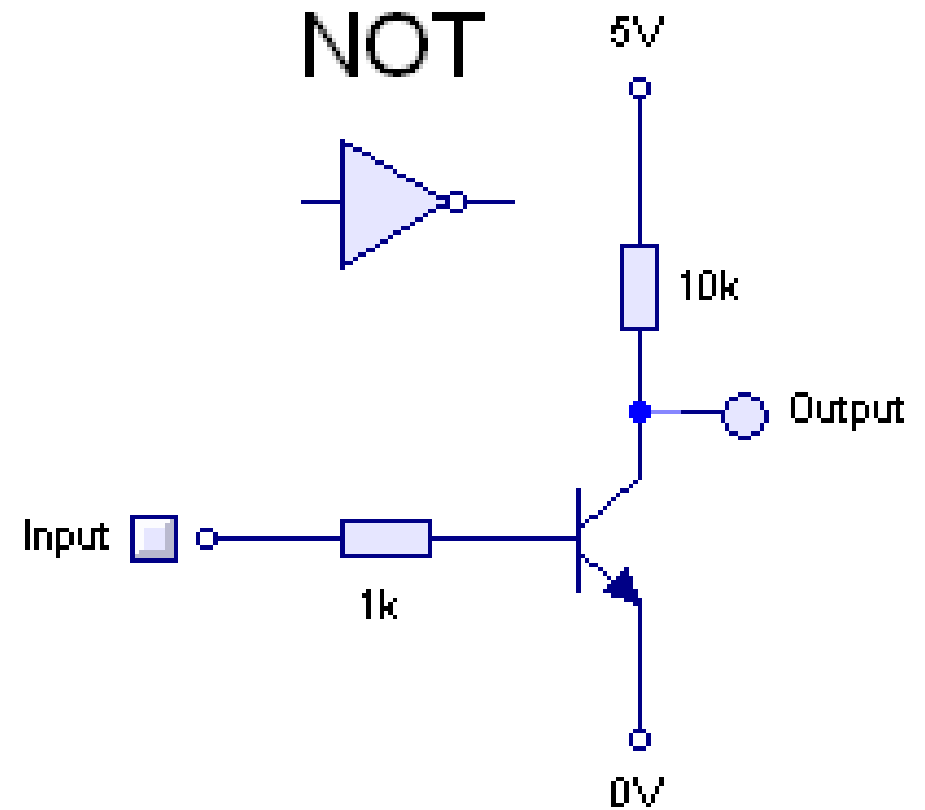
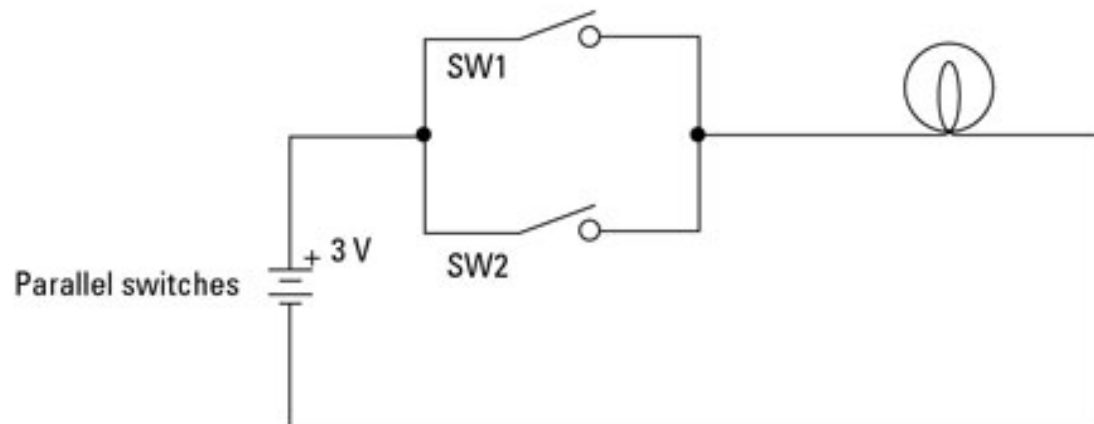
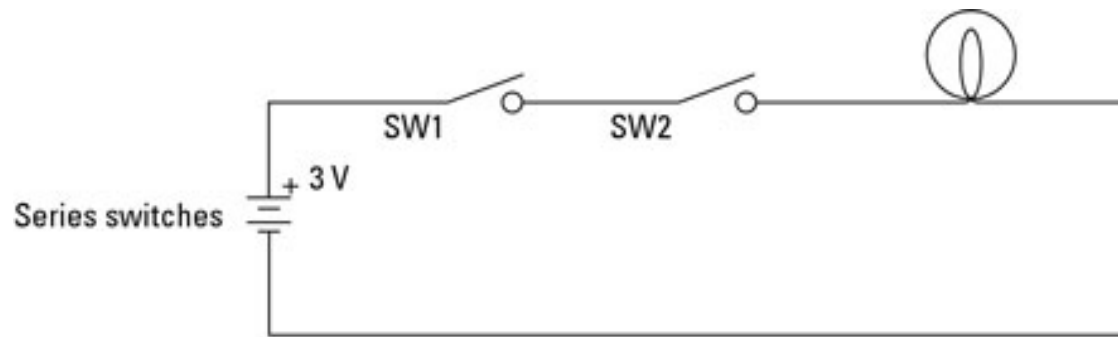
Transistor



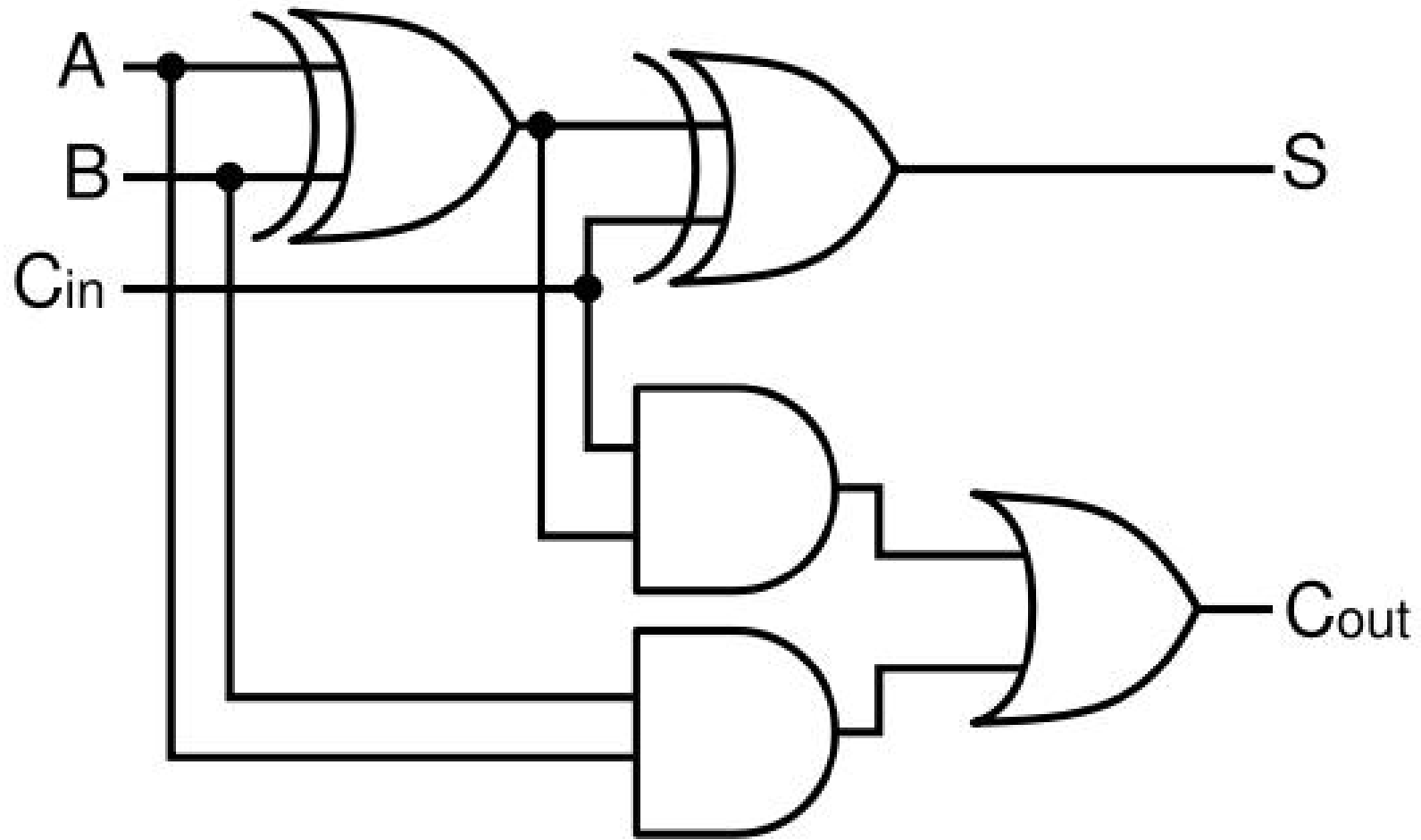
Värvad



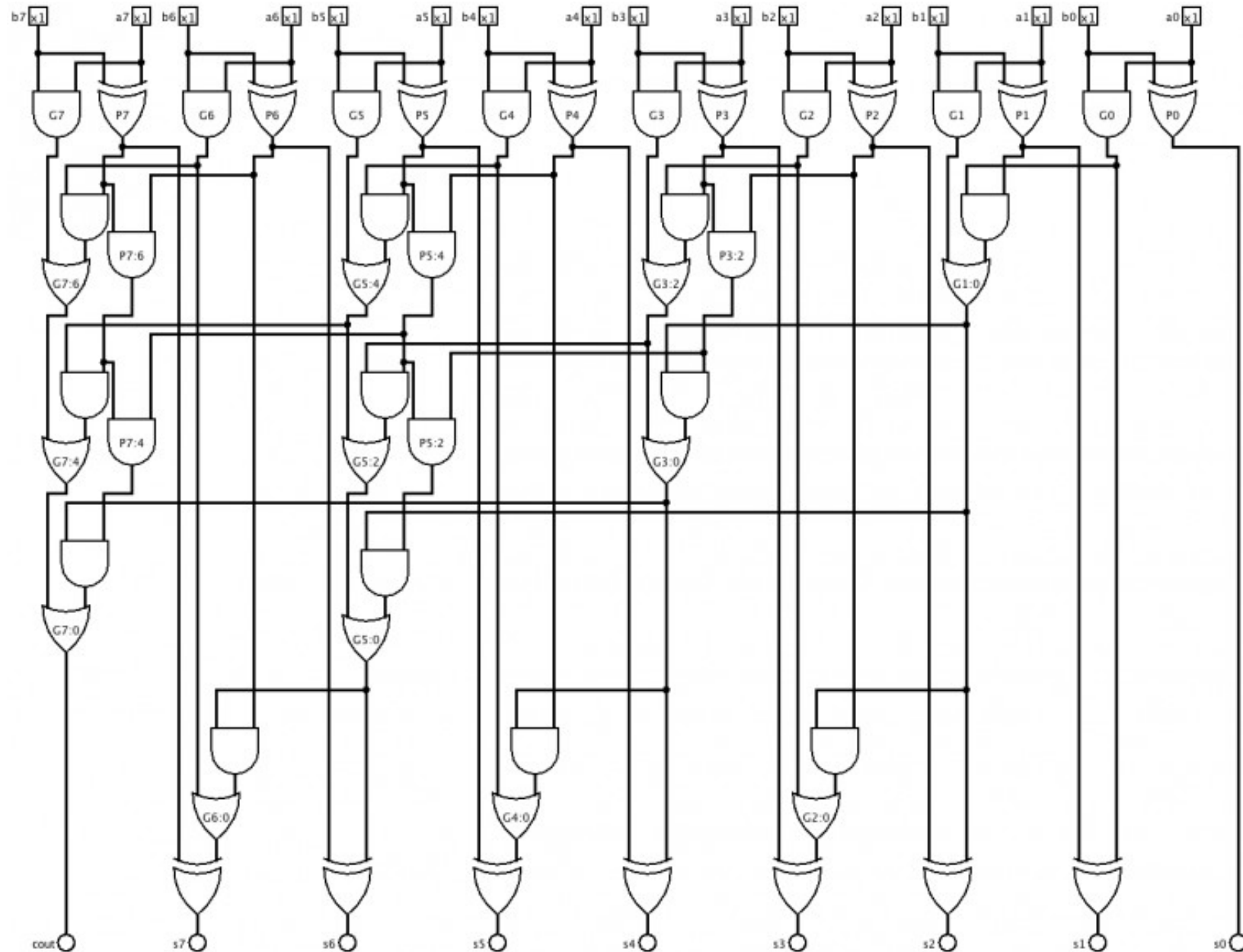
Väravad



Skeemid



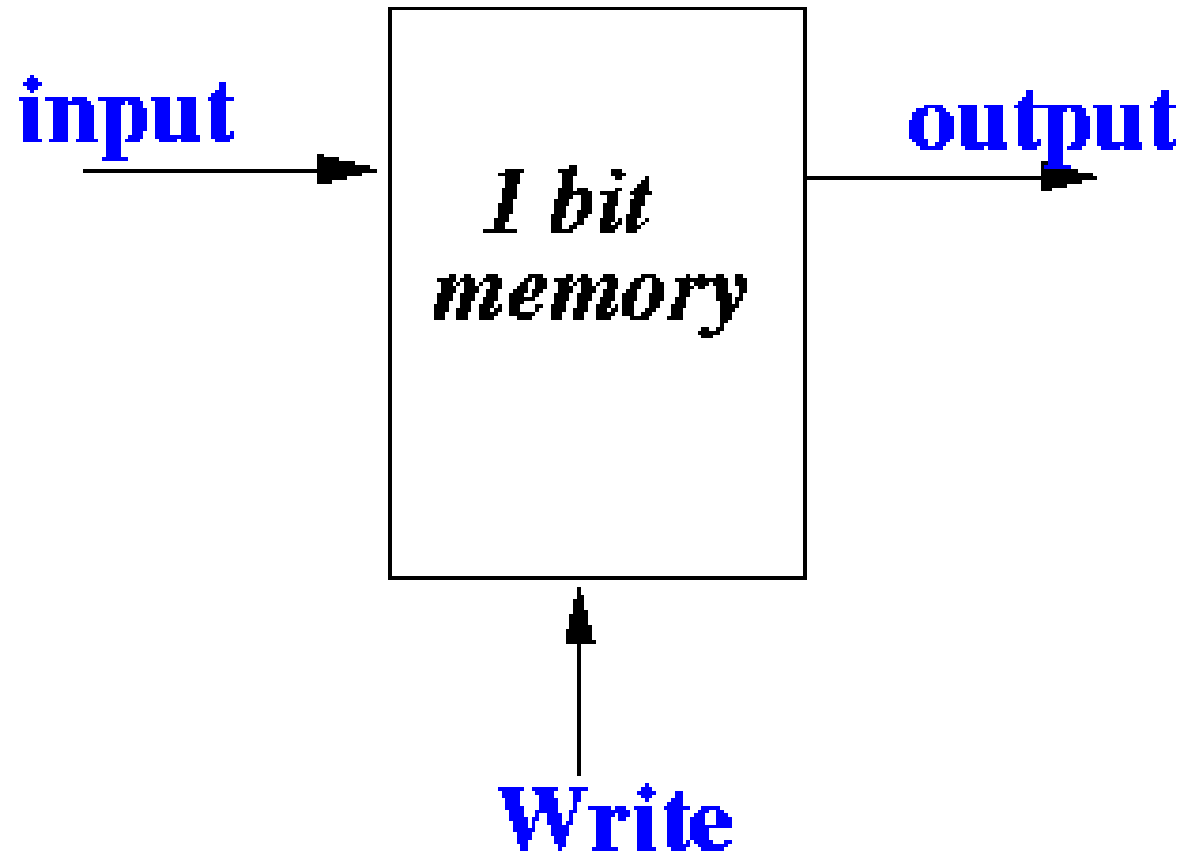
Skeemid



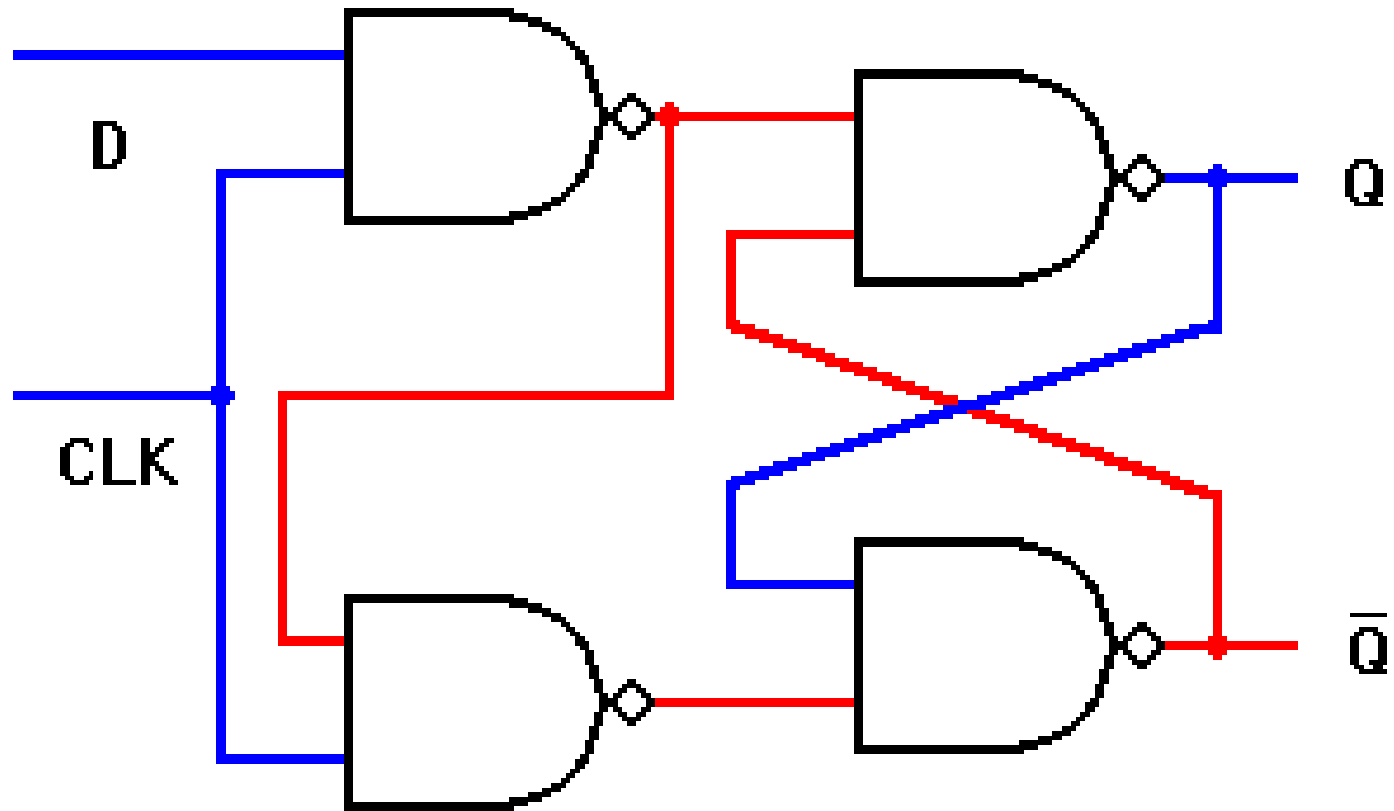
Kalkulaator



Mälu



Mälu



Registrid

- A
- B
- C

Pane A-sse 1 ja B-sse 2

Liida A ja B ja pane tulemus A-sse

Kopeeri A väärtus C-sse

jne

Jada

OP1 -> OP2 -> OP3 -> ...

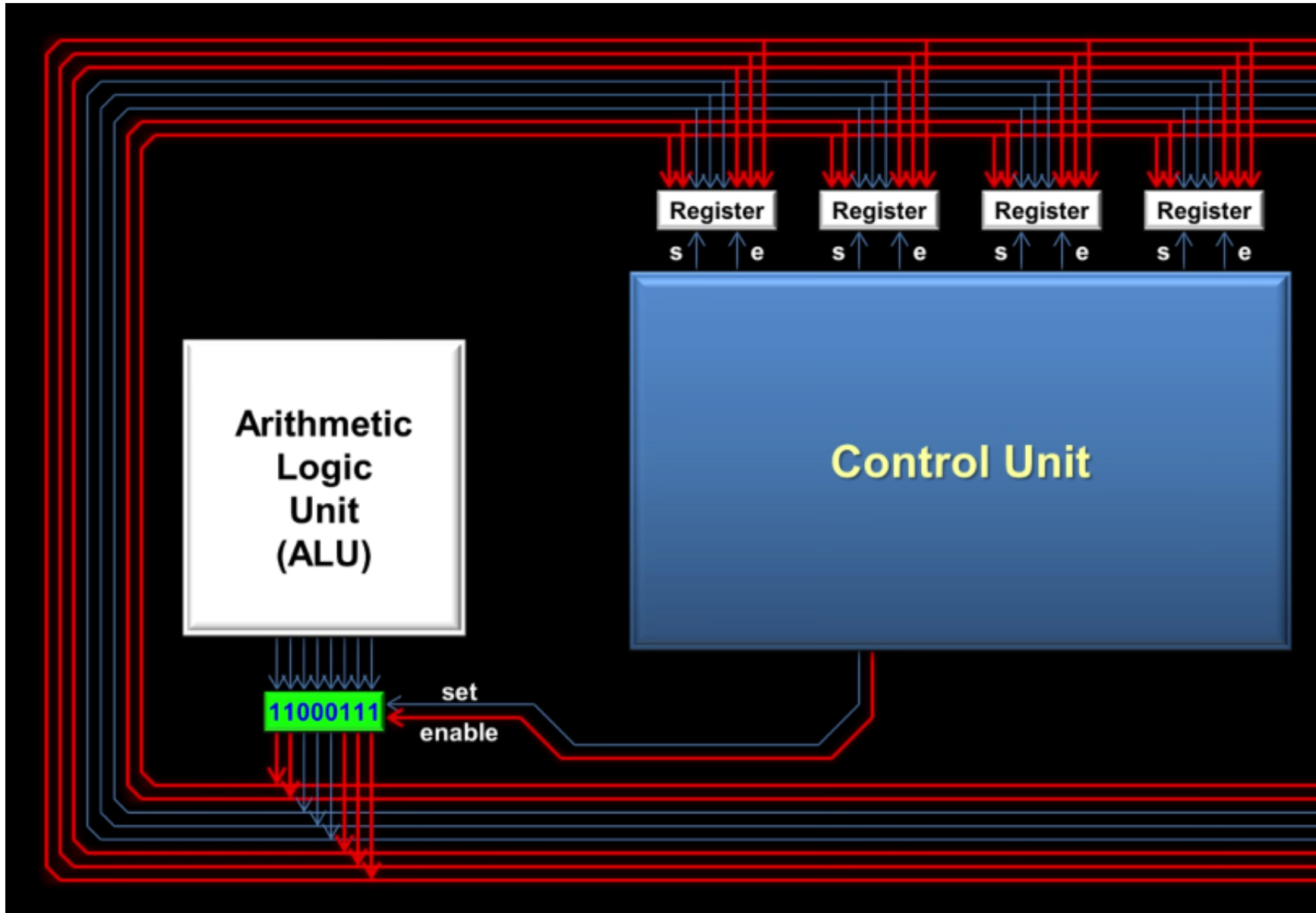
42

+3

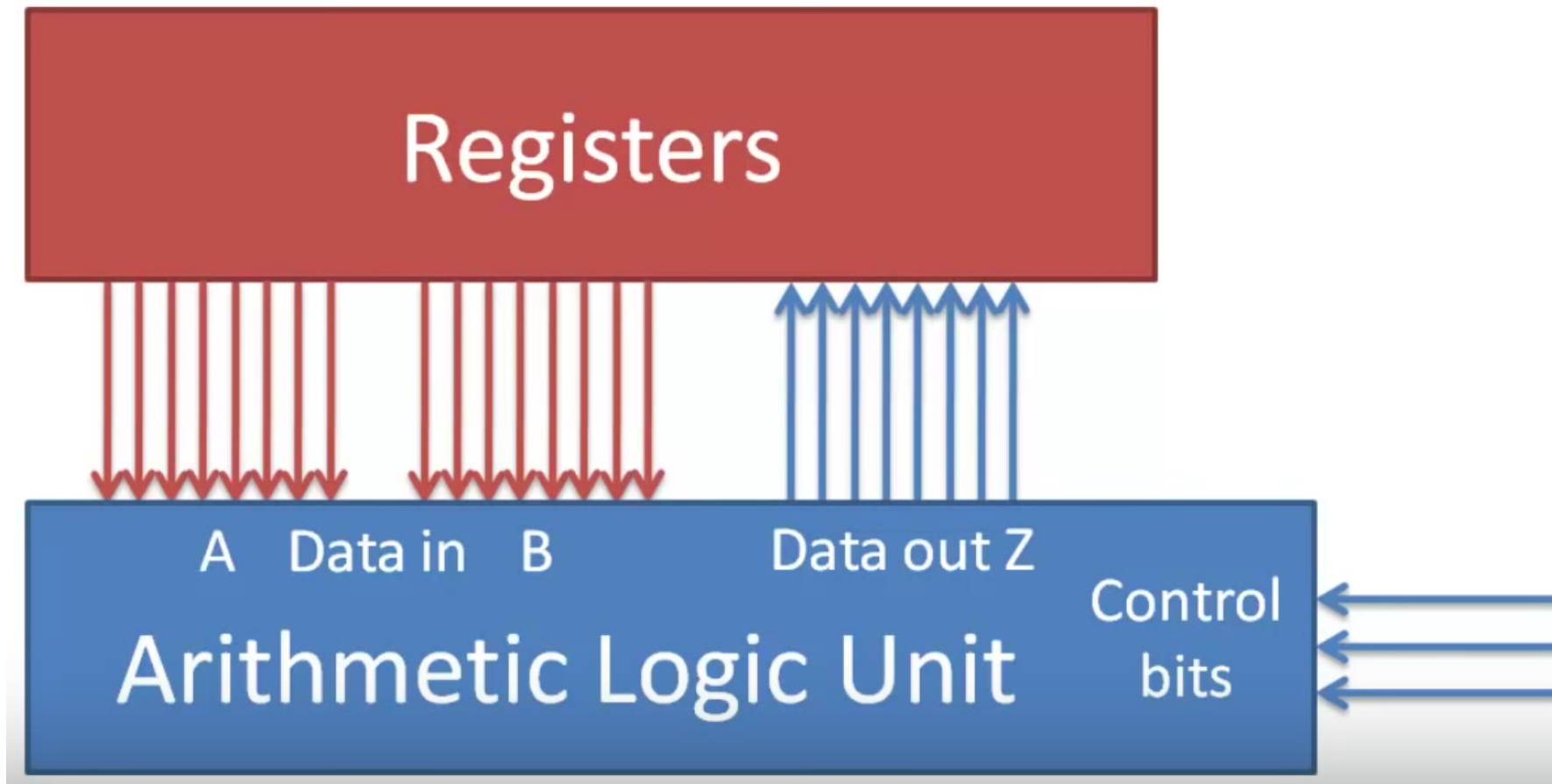
-2

...

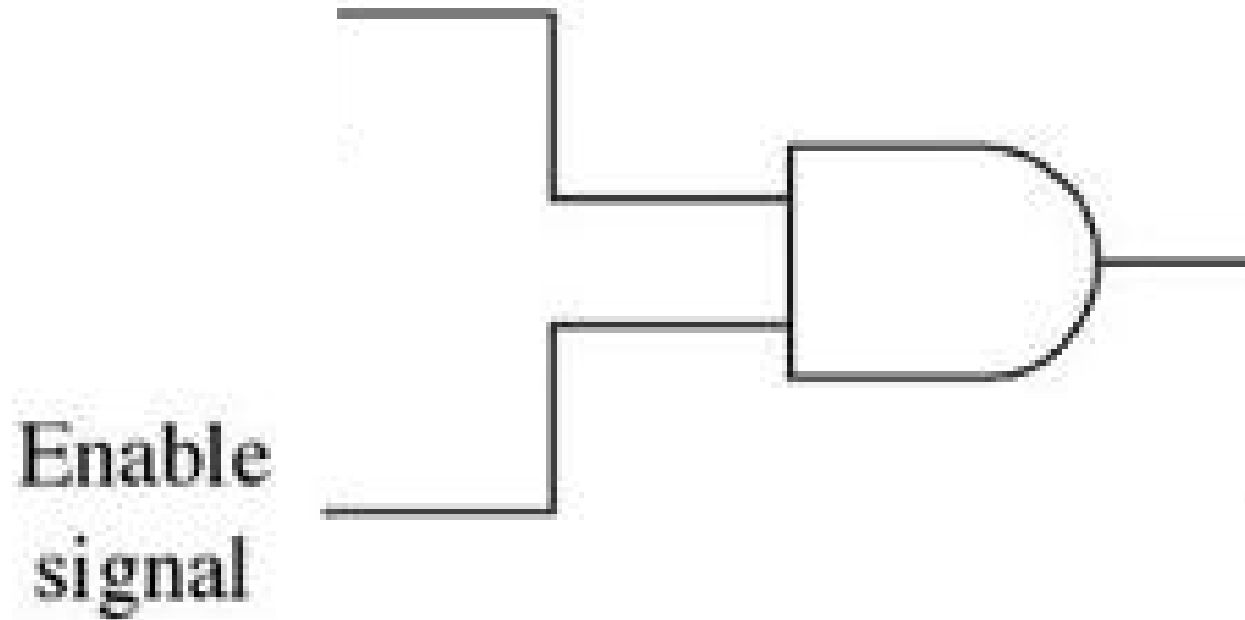
CPU



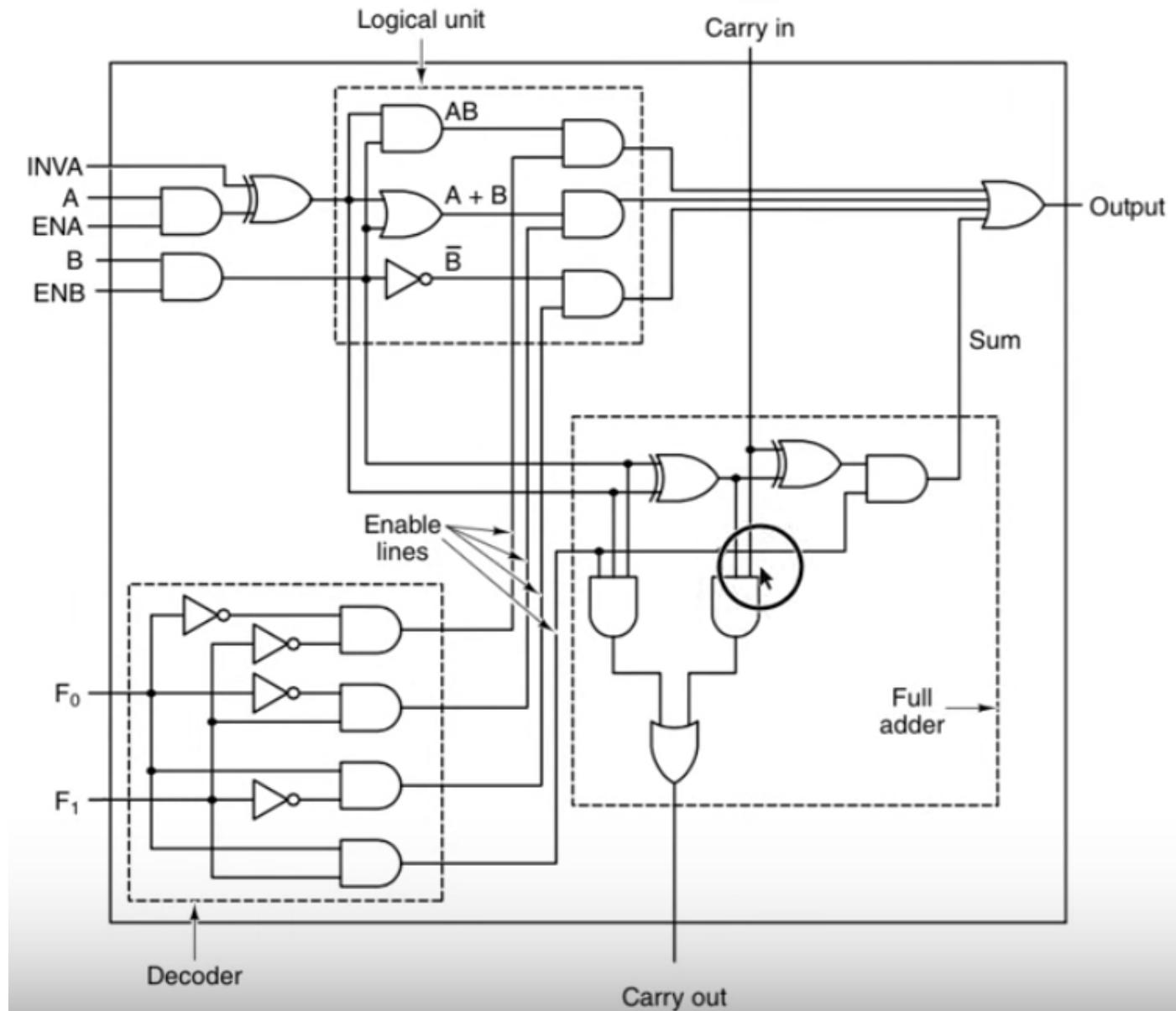
ALU Operatsiooni valimine



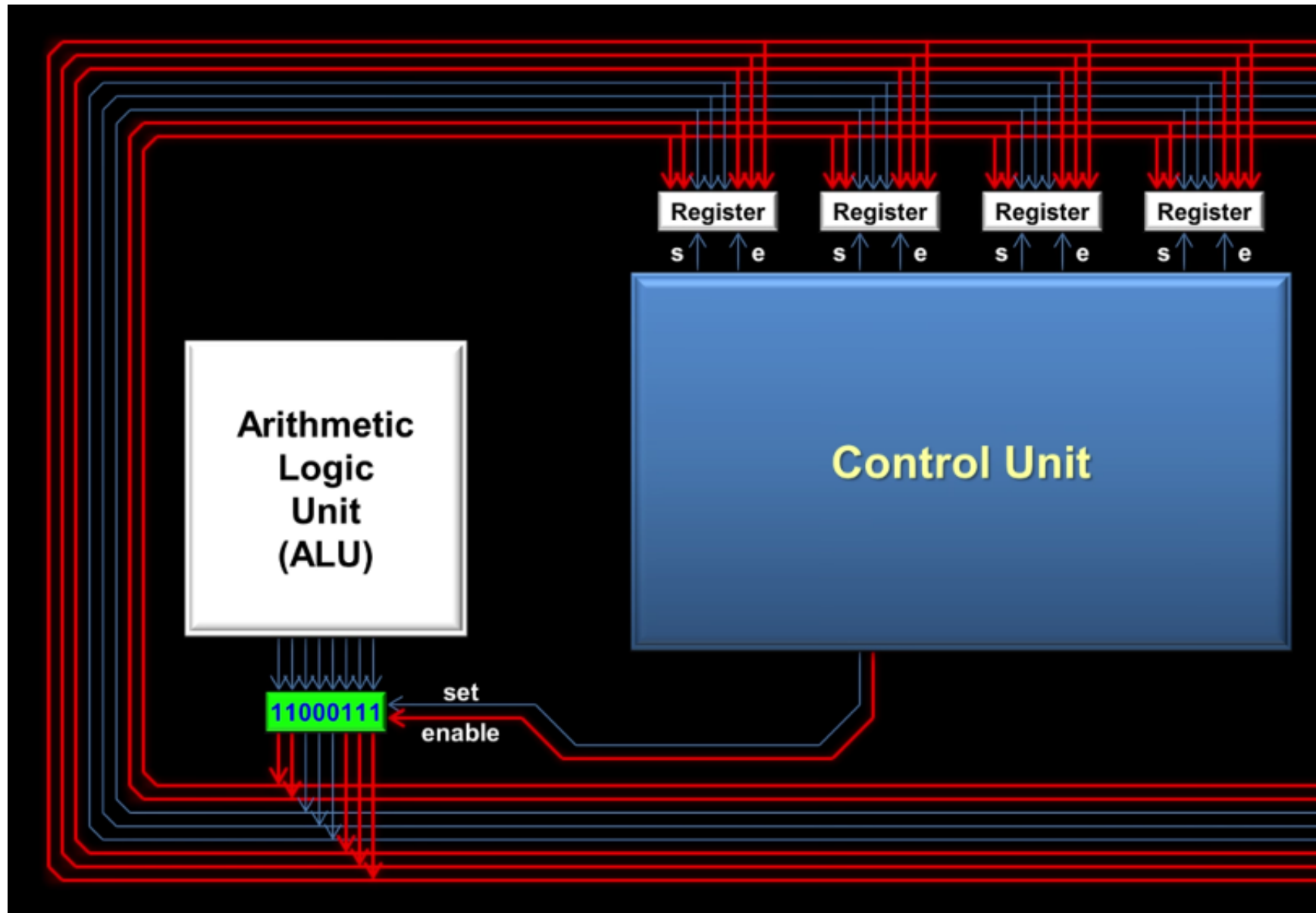
Enable ja Set



ALU Operatsiooni valimine



Control Unit



Control Unit

- Sisend: operatsiooni kood
- Väljund: Set, Enable, jne signaalid

Operatsioonide edasi liikumine

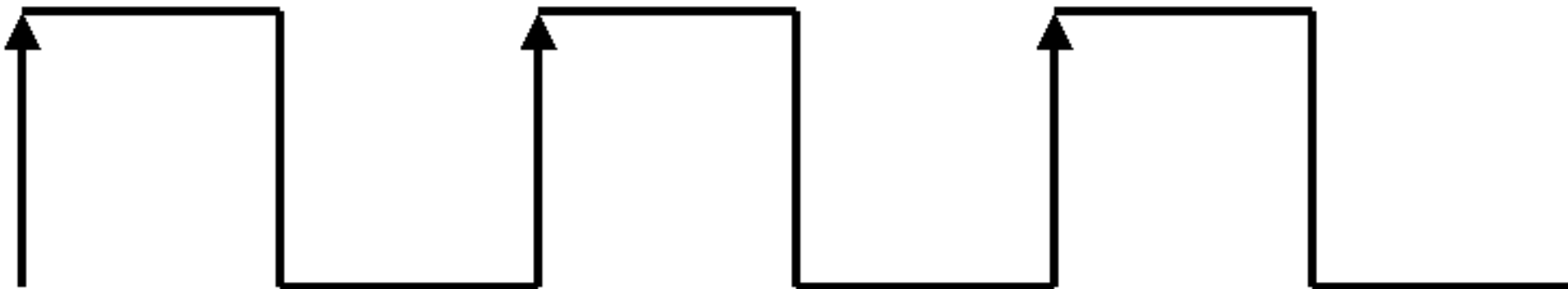


Clock



~15% ühendustest

~20 % energiast



Registrid

A	B	C	D	IP	SP
00	00	00	00	00	E7

Käsud

- Lae registrisse A arv 42

A	B	C	D	IP	SP
00	00	00	00	00	E7

06 00 2A 00 00 00 00 00 00 00 00 00 00 00 00 00

A	B	C	D	IP	SP
2A	00	00	00	03	E7

06 00 2A 00 00 00 00 00 00 00 00 00 00 00 00 00

Käsud

- Lae registrise A arv 1
- Liida registri A väärtusele 2
- Liida registri B väärtusele 7

A	B	C	D	IP	SP
03	07	00	00	09	E7

06	00	01	0D	00	02	0D	01	07	00	00	00	00	00	00	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Assembler

A	B	C	D	IP	SP
03	07	00	00	09	E7

06 00 01 0D 00 02 0D 01 07 00 00 00 00 00 00 00

0600010D00020D0107

MOV A, 1
ADD A, 2
ADD B, 7

Simulaator

- <https://schweigi.github.io/assembler-simulator/>

MOV

MOV reg, reg

MOV reg, address

MOV reg, constant

...

INC

INC reg

DEC reg

ADD, SUB

ADD reg, reg

ADD reg, address

ADD reg, constant

SUB reg, reg

SUB reg, address

SUB reg, constant

MUL, DIV (reg A)

MUL reg

MUL address

MUL constant

DIV reg

Hüpped

JMP address

JZ address

JNZ address

...

Märgendid (label)

JMP sub
HLT

sub:
MOV A, 2

A	B	C	D	IP	SP
00	00	00	00	00	E7

1F	03	00	06	00	02	00	00	00	00	00	00	00	00	00	00
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Lipud

A	B	C	D	IP	SP	Z	C
00	00	00	00	00	E7	FALSE	FALSE

```
MOV A, 2 ; Z = false
DEC A ; Z = false
JZ sub
DEC A ; Z = true
JZ sub
```

PUSH, POP

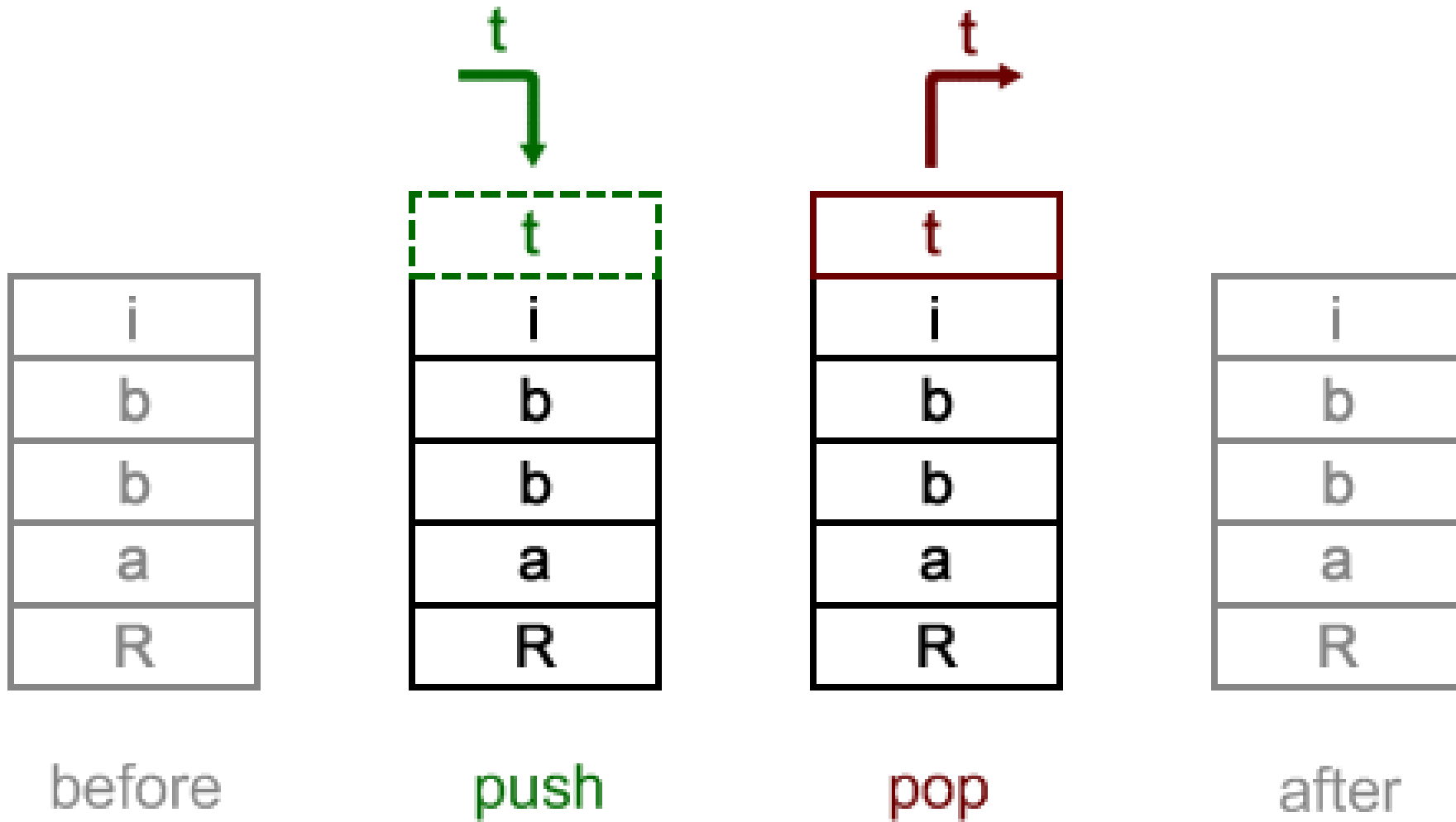
PUSH reg

PUSH address

PUSH constant

POP reg

Stack



Väljund

CALL

- CALL address
- RET

Fibonacci jada

1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ...

Fibonacci jada

	A	B	Stack
MOV A, 2	2	0	[]
MOV B, 3	2	3	[]
PUSH B	2	3	[3]
ADD B, A	2	5	[3]
POP A	3	5	[]

Assembler praktikas

Mitu lõime

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

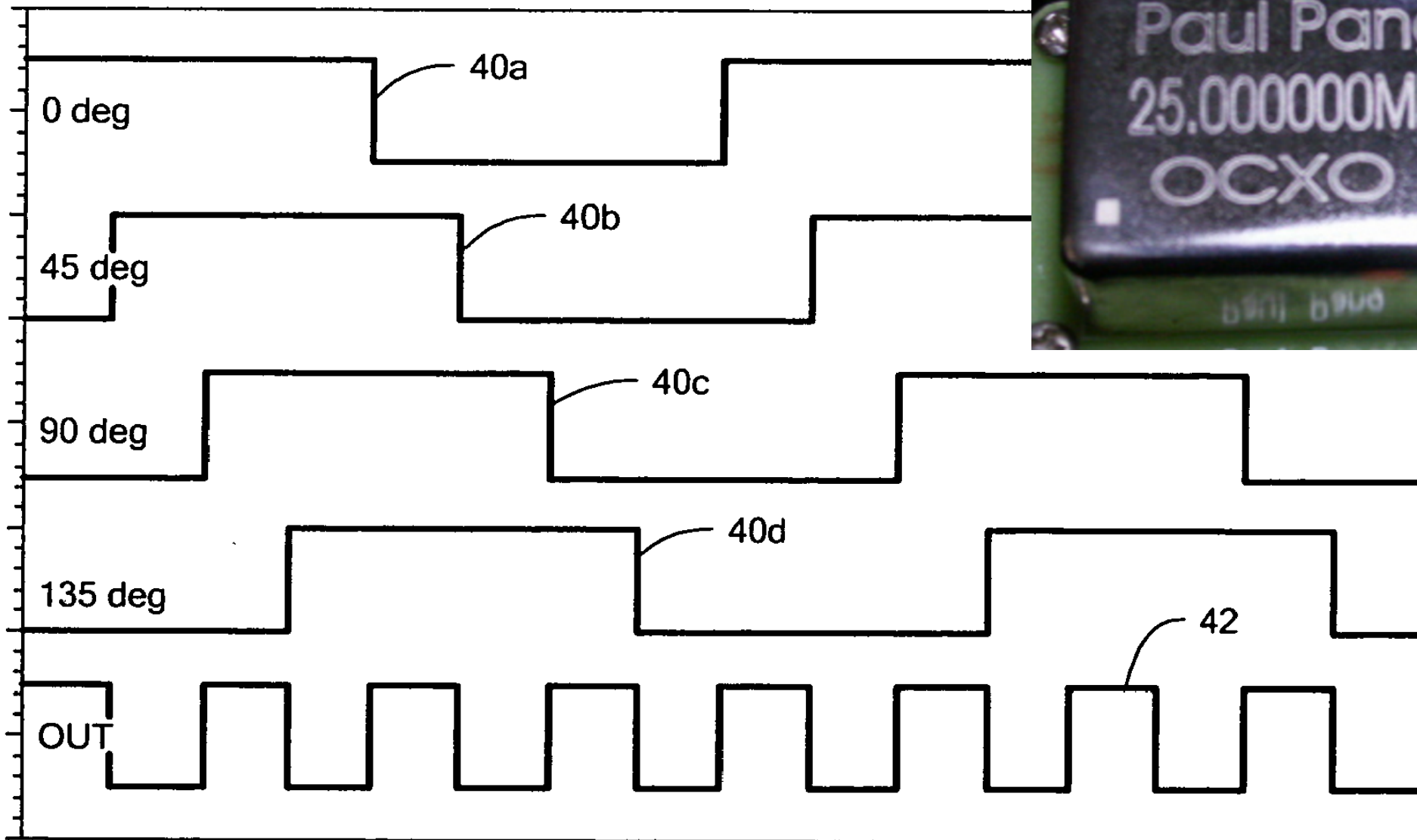
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

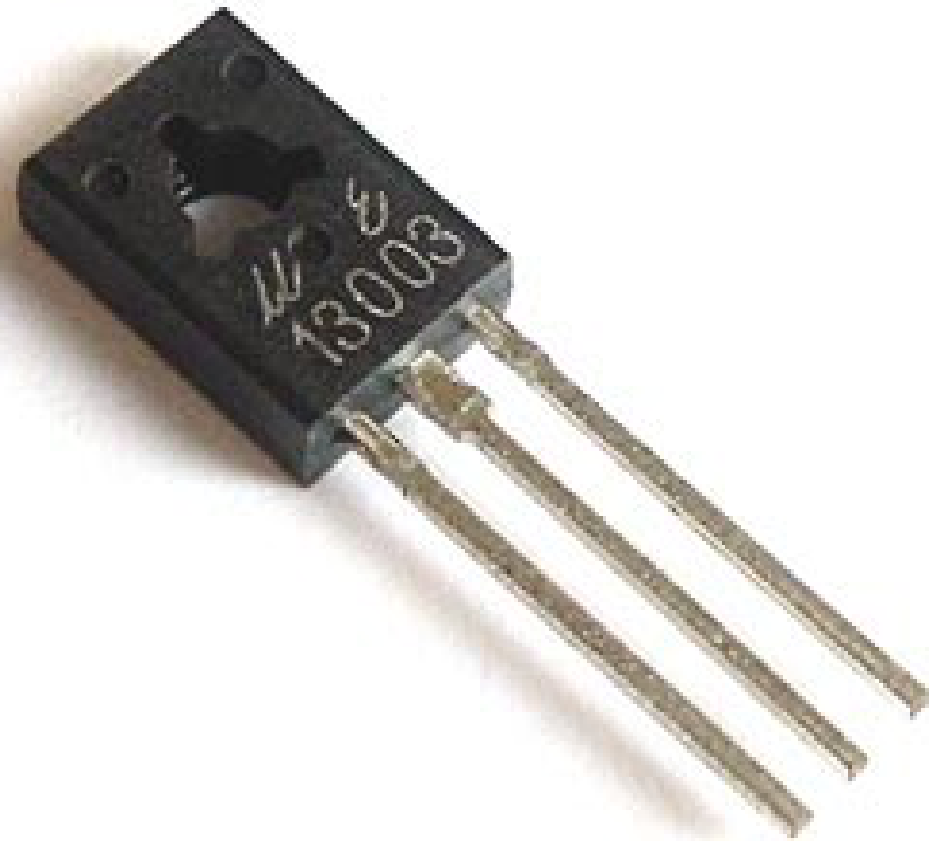
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

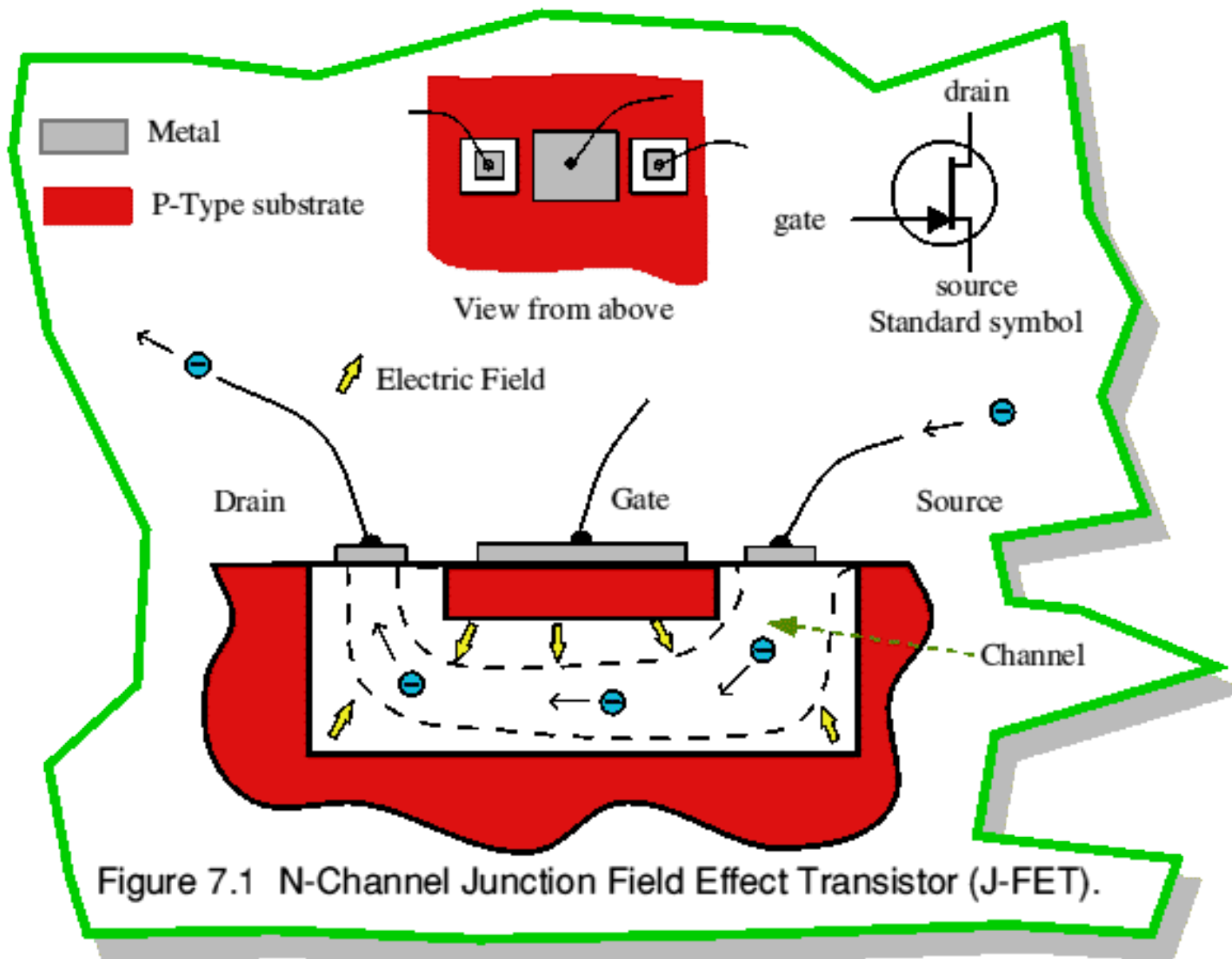
Kiirusest



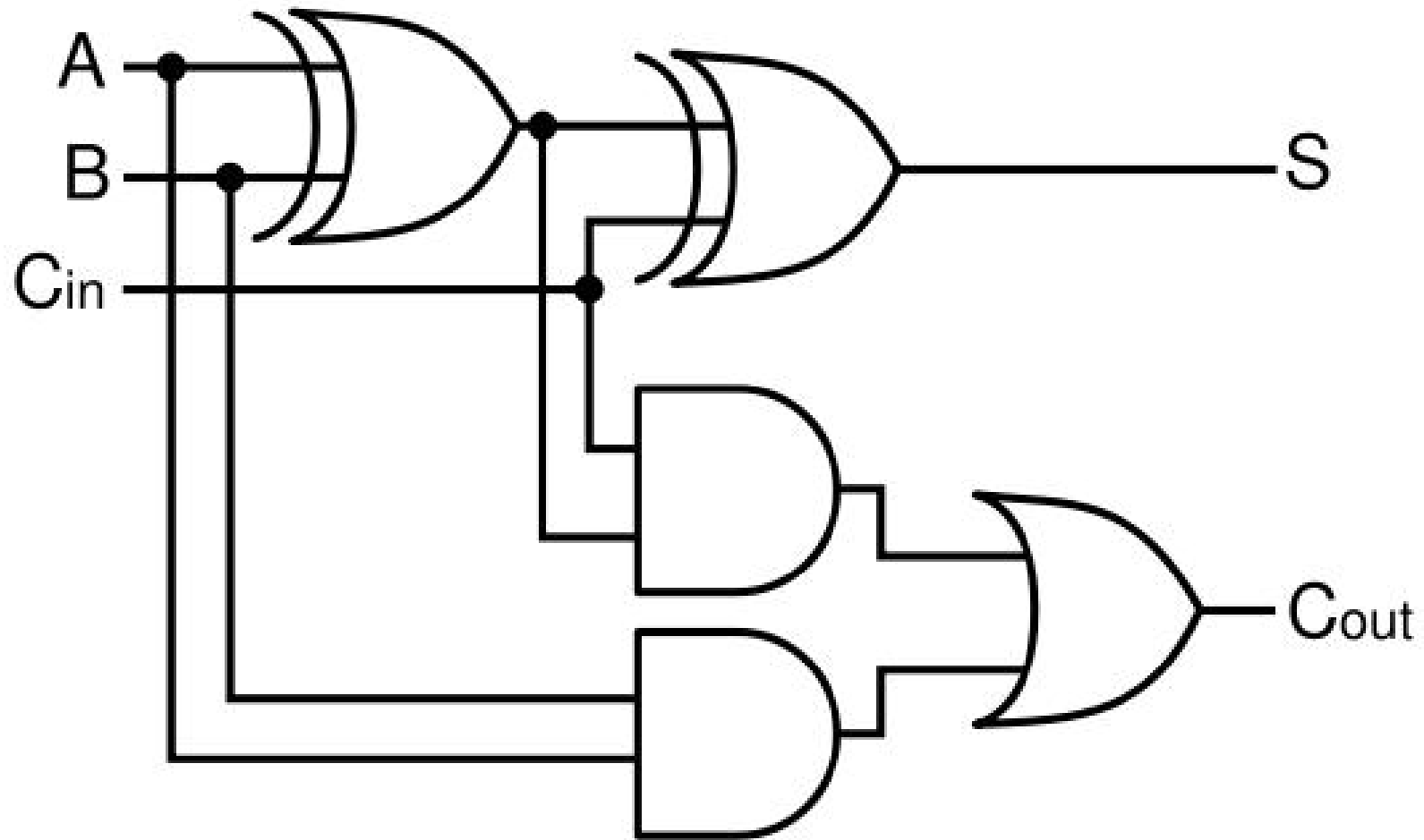
Transistor



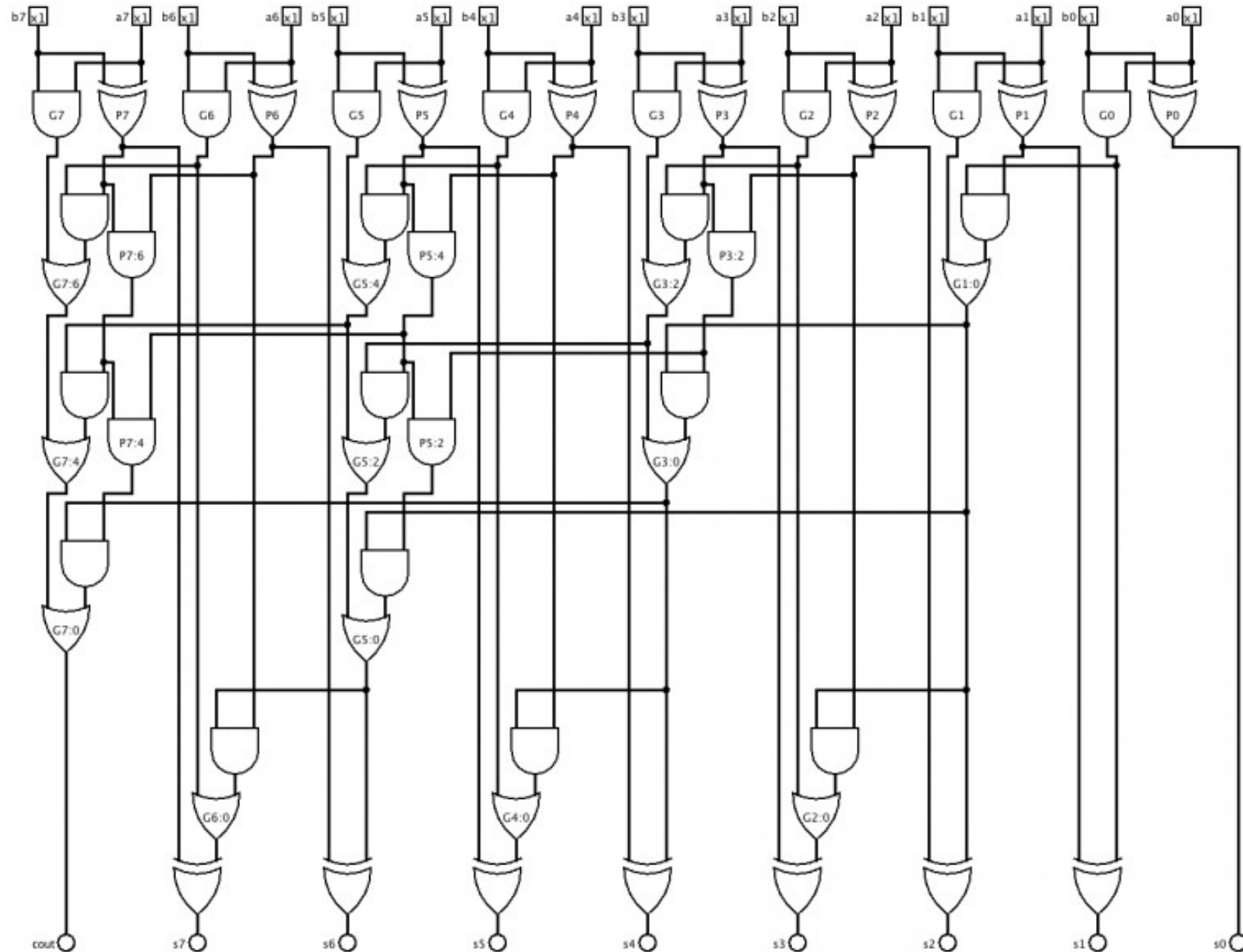
Transistor



1 bit adder



8 bit adder



Kiirusest

- Tõsta sagedust
- Tõsta pinget