

• •

« , »

« , »

.....	2
.....	3
1.	
'	()..... 5
1.1.	' .. 5
1.1.1	' ... 8
1.1.2 8
1.2.	SIMD 12
1.2.1	- 12
1.2.2 17
1.2.3 22
1.3. 33
1.3.1 33
1.3.2	' 34
	1 40
2	
.....	41
2.1	' .. 41
2.1.1	' IMD 41
2.1.2	SMP 42
2.1.3 50
2.1.4	()..... 59
2.1.5	' 67
2.2 71
2.2.1 71
2.2.2	' 75
2.2.3 78
2.2.4 87
2.2.5 93
2.3. 98
2.3.1 98
2.3.2	' 102
	2 117
3. 118
3.1. 118
3.1.1 118
3.1.2 122
3.2. 126
3.2.1 126

3.2.2	128
3.2.3	130
3.2.4	133
3.3	137
3.3.1	137
3.3.2	139
3.3.2	141
3	143
4.	'	
	- 144
4.1.	- () 144
4.1.1	' - 144
4.1.2	- 147
4.1.3	- 152
4.2.	' 153
4.2.1	' 153
4.2.2	' 156
4.2.3	' 161
4.2.4	- ' 163
	4 167
5.	 168
5.1.	 168
5.1.1	 168
5.1.2	C 170
5.2.	 176
5.2.1	 176
5.2.2	 176
5.2.3	 177
5.2.4	 179
5.3.	 180
5.3.1	 180
5.3.2	 180
5.3.3	 184
	5 186
	 187

COMA – (Cache Only Memory Architecture);

DSM – (Distribute Shared Momory);

HDD – (Hard Disk Drive);

JBOD – (Just A Bunch Of Disks);

NORMA – (No Remote Memory Access);

NUMA – (Non-Uniform Memory Access);

SISD – (Single Instruction, Single Data);

SIMD – (Single Instruction, Multiple Data);

MISD - (Multiple Instruction, Single Data);

MIMD – (Multiple Instruction, Multiple Data);

SMP – (Symmetric Multi Processors);

MPP – (Massively Parallel Processing);

RISC – (Restricted (reduced) instruction set computer);

RAID – (Redundant Arrays of Inexpensive Disks)

– ;

– ;

– ;

– ;

– (-);

– ;

1.

, ()

1.1.

,

1.1.1

,

1833

,

«

»,

100

, ,
1945 - 47 .

, ,
1947 .

,

-

.

,

,

.

,

.

,

,

.

.

-

.

,

.

,

.

,

,

,

.

(CPU, Central Processor Unit -),

(. chip).

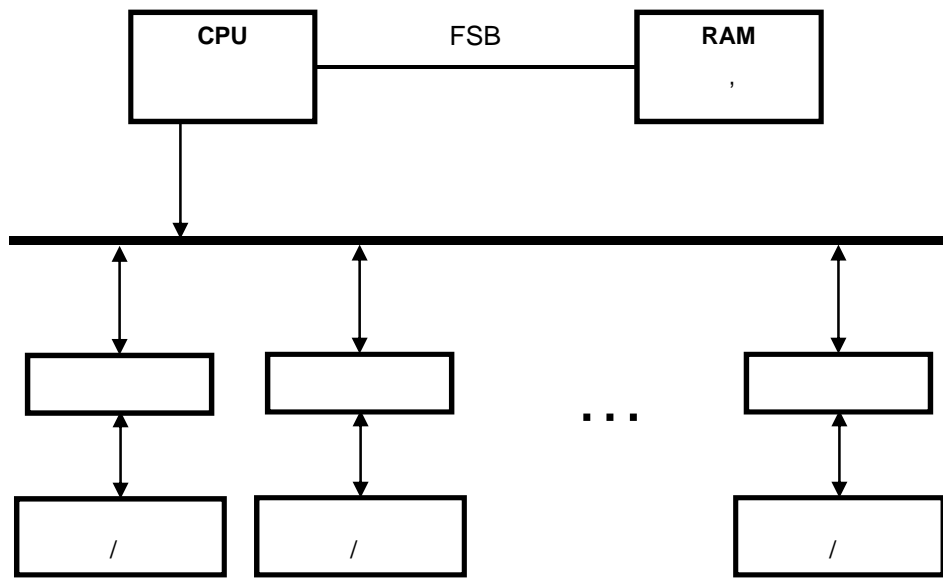
()

(,)

/).

2N, N -

(.1.1).



.1.1

(,) - ()

2000-2005

« »

MIMD (Multiple Instruction, Multiple Data) –

;

(
MIMD.

1.3).

79

1024,

1024,

79.



.1.3

1.4).

79



. 1.4

Symmetric Multi Processors (SMP). SMP

SMP

: Symmetric Multi Processors Shared Memory Processors.

1.2. SIMD

1.2.1

SIMD-

SIMD

GFLOPS.

(),

. SIMD-

:

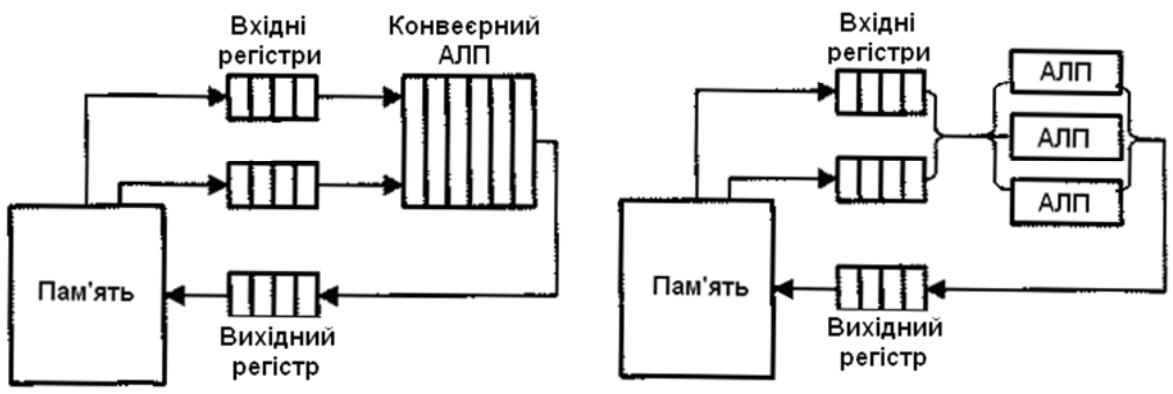
()),

().

:

- ;
- ;
-

. 1.4.



a)

)

. 1.5

:

(. 1.5,)

().

(. 1.6,).

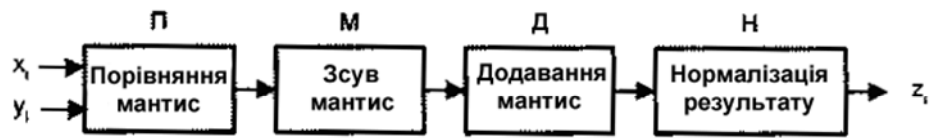
(. 1.6,).

(. 1.6,).

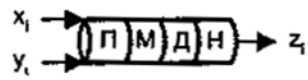
(. . 1.5,).

. 1.6, .

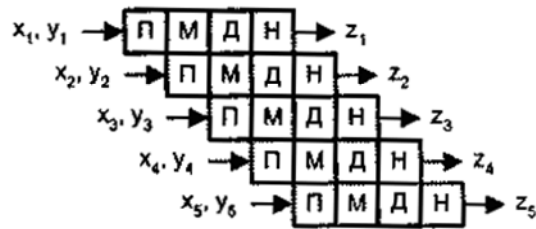
. 1.6, .



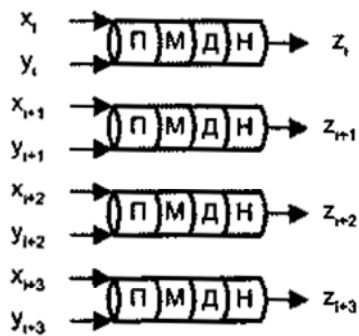
а)



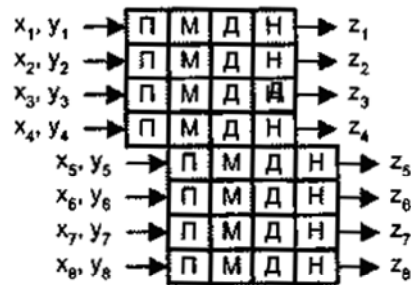
б)



в)



г)



д)

. 1.6

:

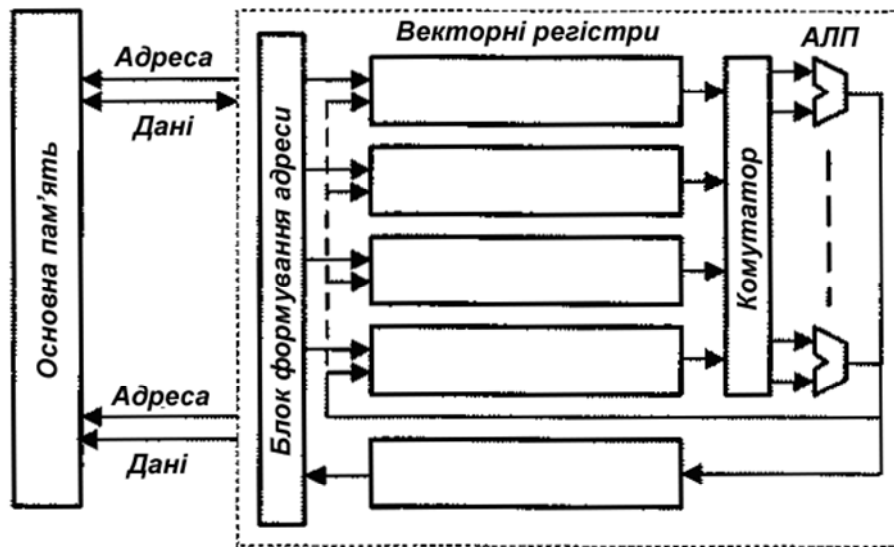
;

; —

;

;

. 1.7.



.1.7

()

X/Y

$X(I/Y)$

FIFO,

50-100

$(V_a, V_b, V_c \dots)$

, , ;
 • , ;
 • , .
 , .
 « - » ,
 .
 « - » , .
 , 64 .
 .
 « ' - ' »
 , « - »
 .
 , « ' - ' »
 , ,
 , ,
 , .
 « ' - ' » , ,
 .
 , .
 :

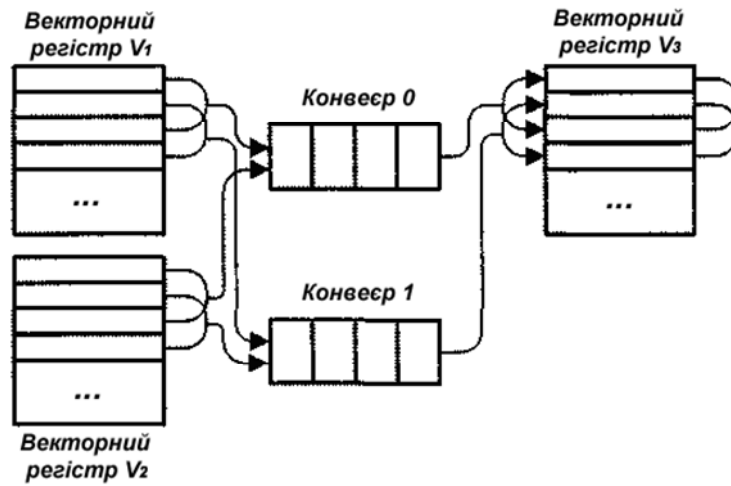
$$T = s + x N \quad (1.1)$$

s — , — , (0.5, 1
 2) N , — .

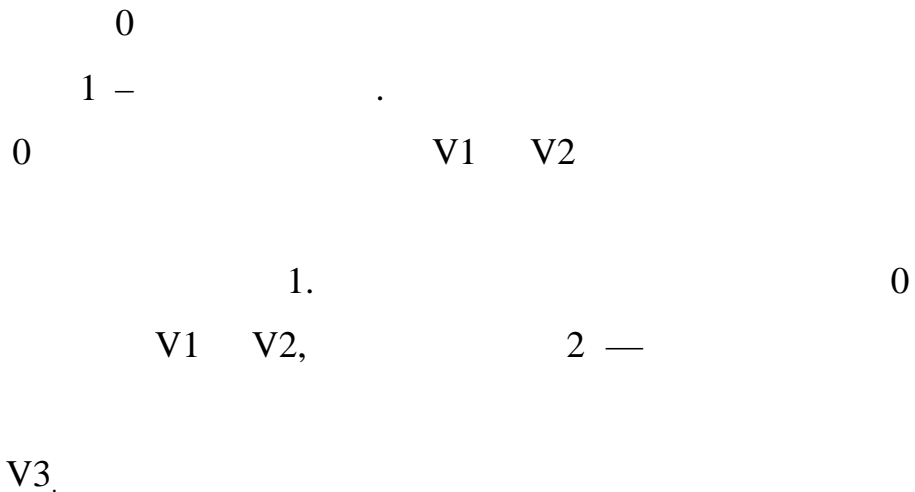
()

Cray C90,

(.1.8).



. 1.8.



1.2.2

(array processor),



. 1.9.

SIMD-

()

(front-end computer).

SIMD-

SIMD-

(. 1.10).

«

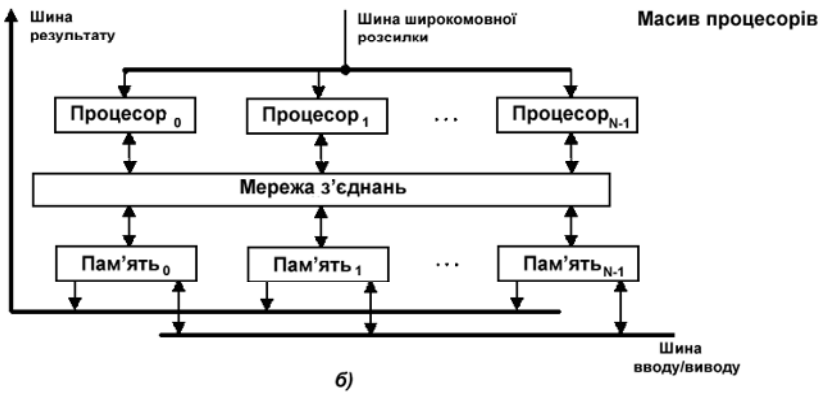
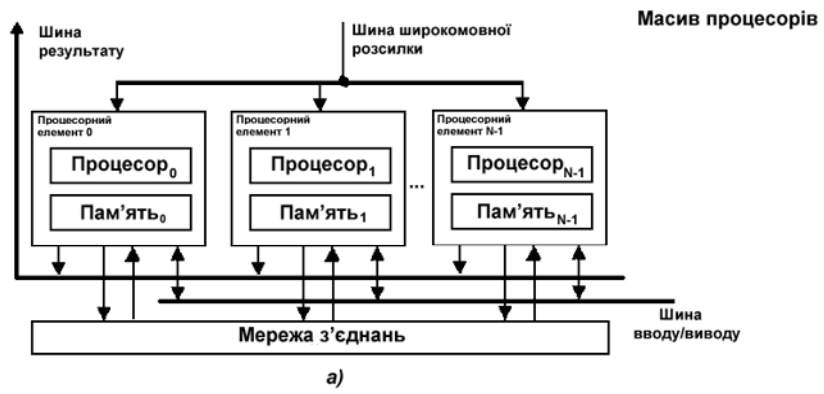
» (« - »), N

()

(. 1.10,).

—

MasPar MP-1, Connection Machine CM-2, GF11, DAP, , STARAN, ILLIAC.



. 1.10.

, — « — »;

— « — » —

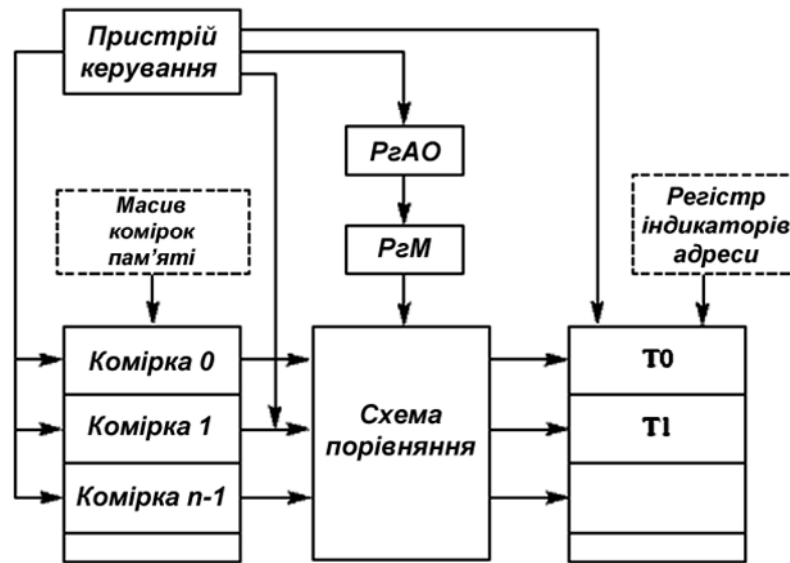
.1.10, .

N

Burroughs

Scientific Processor (BSP), Texas Reconfigurable Array Computer TRAC.

(SIMD = Single Instruction Multiple Data).



. 1.11

(. 1.11)

, : " — " " —
 " " — " " —
 " " — " " —
 " . . .
 .
 ,
 ,
 SIMD (STARAN, PEPE) MIMD. (

STARAN).

(32 —

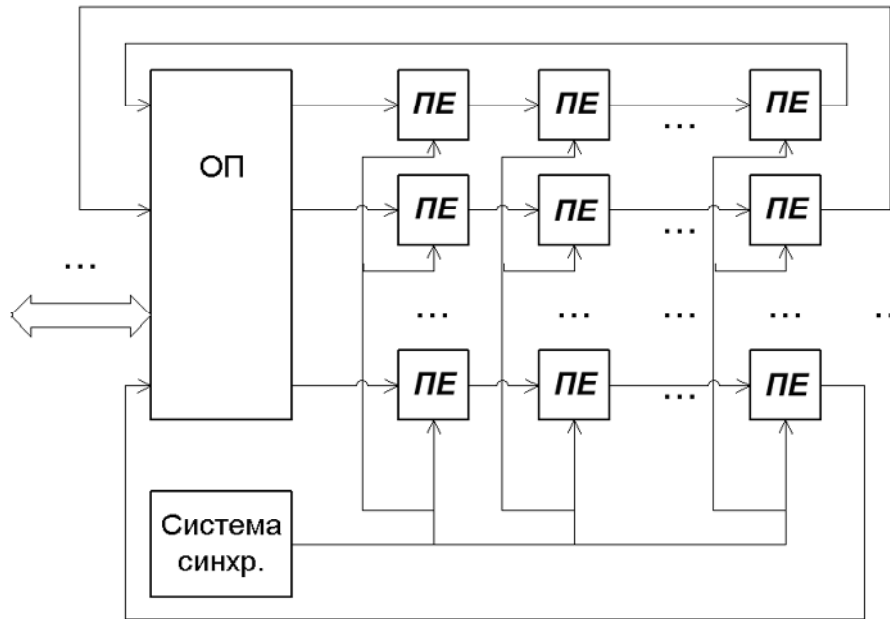
()
 « » (), '
 .
 .
 ().
 ,
 .
 « » (systolic),
 (— ' , ,
 , « »
).
 ,
 .

MISD,

MIMD-

MIMD

. 1.12.



. 1.12

- ;
- , ,
- ;
- ,
- ;
- ;
- ,
- ;
- ,
- ;
- ().

(VLIW)

(VLIW, Very Long Instruction Word)

80-

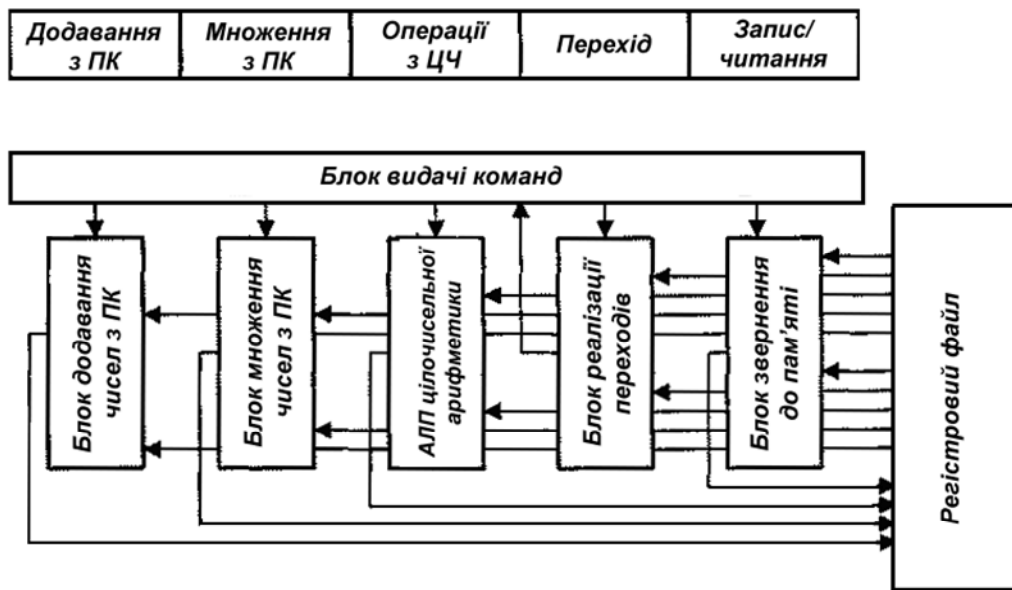
—

VLIW

. VLIW

« »

() ;



.1.13

256 1024

),

. 1.13

VLIW-

. ,
 , . ,
 ,
 VLIW- , ,
 .
 , ,
 RISC- , VLIW
 RIS - .
 3 20. ,
 .
 ,
 (, . .),
 .
 1 . / VLIW-
 VLIW
 .
 VLIW — ,
 .
 ,
 , , ,
 , , .
 , ,
 , , .

, ()
.
- . -
.
, - ,
.
- . -
.
, - ,
(-). , '
- , - ,
.
, , . -
() . .
1020 ,
.
- ,
.

1994 ,
Southern California

Leonard Adleman University of

," "
."
, "
,"
- ,
,

2001

"Nature",

99,8%.

2003

"Proceedings of the National Academy of Sciences"

Weizmann Institute of Science ()

50

330

1.3.2

y_{m1}

()

()

$= \{ , \dots, m \}$,

$Y = \{y_1, \dots,$

$X = \{x_1, \dots, x\}$,

$y = (), = 1, \dots, m.$

$F,$

$Y = F(X)$

$X,$

X.

.1.14.



. 1.14.

X^* Y^* ,
 $= \{ n \}$.
 X^* Y^*
 $Y = F(X^*)$,
 Y^* .
 $Y = F(X)$, X

: 1)

(,); 2)

, , ; 3)

, .

: , M

, 0

.

. = (b... q) -

A, ; S=(s₁,...,s_p) -

, () ; = (1, ...,

r) ; Y=(y₁,..., y_m) —

, S C

Y=F(, S, C); S={S_j} -

; = { j} —

z_α,..., z_ω ∈ (Y ∪ X)

z_α ∈ z_α^{*},..., z_ω ∈ z_ω^{*} z_α^{*},..., z_ω^{*} -

.
= F(Y),

Y=F(, S, C).

: 5 C,

$$\max E = \max_{s \in S, c \in C} \Phi(Y) \tag{1.2}$$

$$: z_{\alpha} \in z_{\alpha}^*, \dots, z_{\omega} \in z_{\omega}^* \tag{1.3}$$

,
X,

S C,

(1.2),

(1.3).

$$Y=F(, S, C),$$

(1.2)

(1.3).

$$= F(),$$

S

S

C

(

)

1.

(

)

2.

:

3.

(

.)

2.1

2.1.1

IMD

SIMD

SIMD-

MIMD. MIMD-

MIMD

MIMD-

()

MIMD-

(tightly coupled),

(SMP, Symmetric Multiprocessor)

(NUMA, Non-Uniform Memory Access).

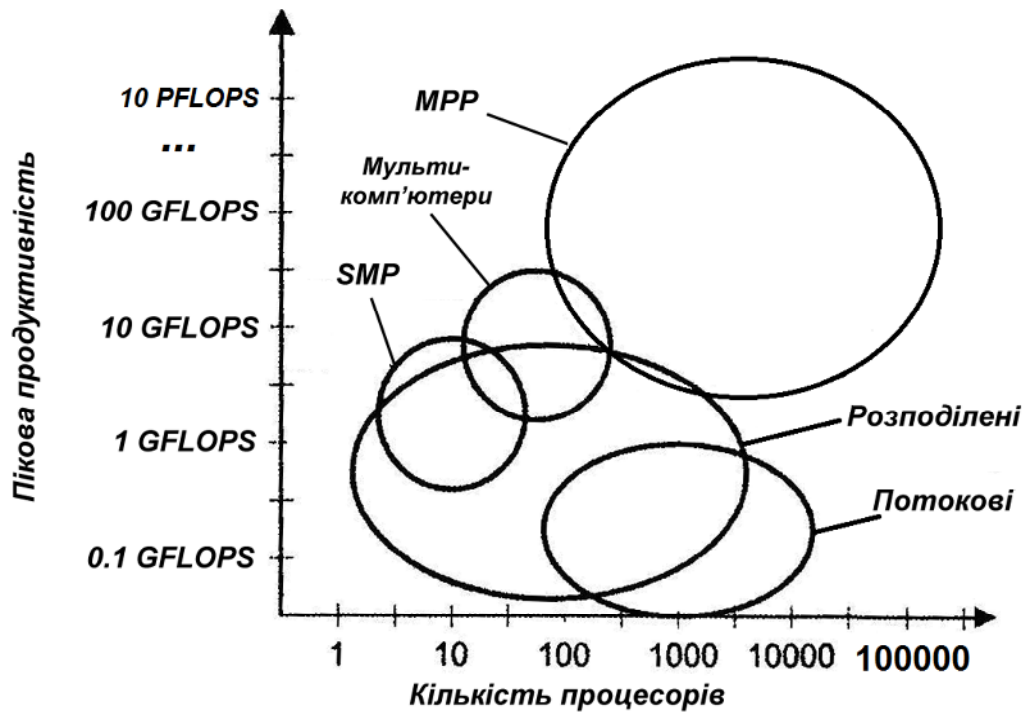
(loosely

coupled)

« »

(, Massively Parallel Processing)

MIMD-



. 2.1

MIMD-

(),

MIMD

. 2.1.

2.1.2

SMP

·
SMP- (SMP,
Symmetric Multiprocessor).

· SMP

:

- ;
- , ;
- , - ;
- /
- ;
- (« »).

-

· SMP

SMP-

(master).

SMP-

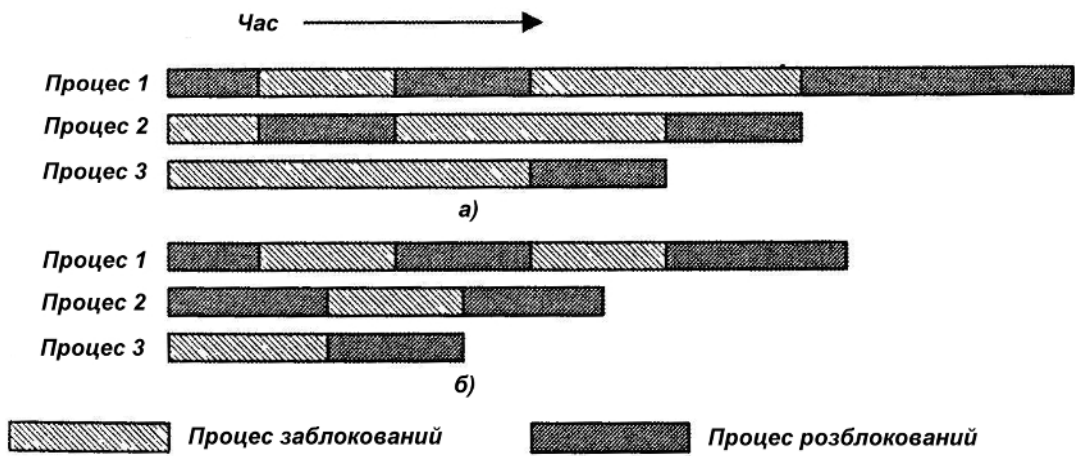
« »

(threads)

SMP-

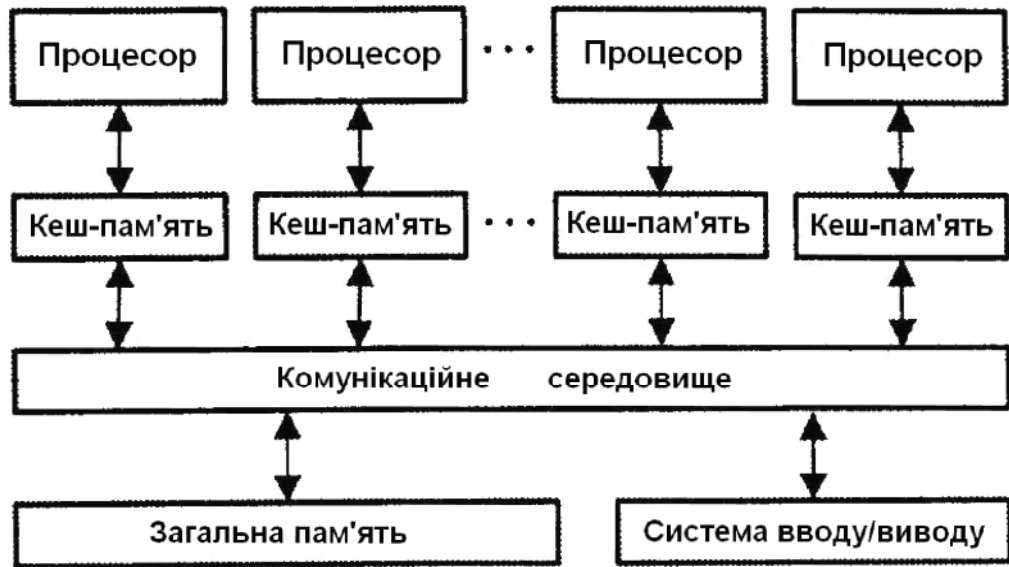
SMP-

(.2.2);



.2.2

2.3



2.3

SMP-

32

RISC-

DEC Alpha, Sun SPARC, MIPS

HP PA-RISC.

SMP-

CISC-

Intel AMD.

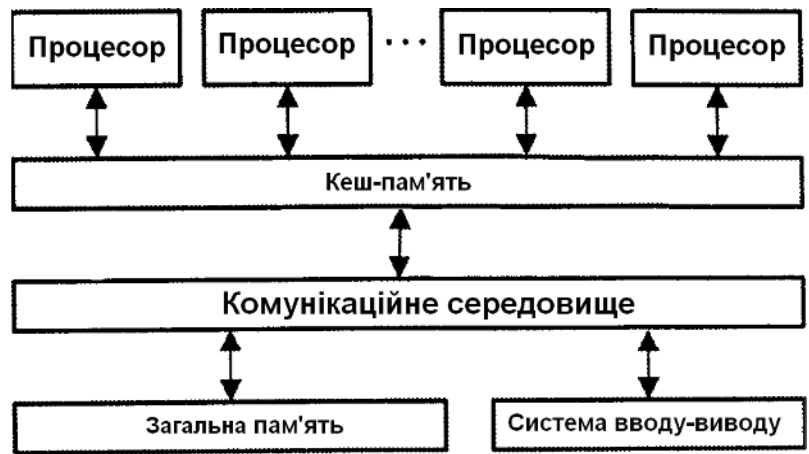
(L1)

(L2)

SMP-

(2.4).

, / .
 , (SMP-
).
 ,
 ,
 ().

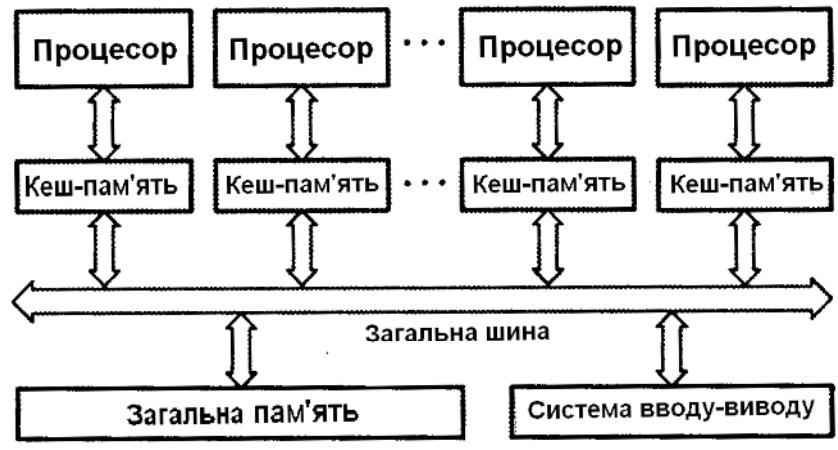


.2.4. SMP-

/).
 SMP- :

- ;
- « »;
- , ;
- .

(.2.5).



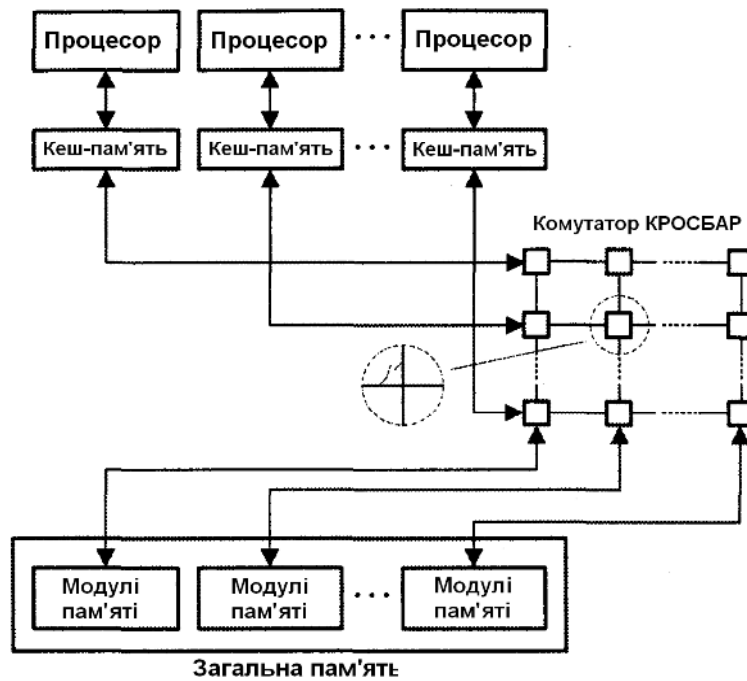
.2.5. SMP-

SMP-

SMP-

Compaq AlphaServer GS140 8400
 Alpha 21264. SMP- HP
 N9000 8 -8500,
 SMP Thin Nodes RS/6000 IBM
 POWER PC 604.
 SMP-
 86. : DELL Power
 Edge, IBM Netfinity, HP NetServer.

« »
 « » (. 2.6)



. 2.6. SMP- « »

SMP-

SMP-

32 64. ,

Enterprise 10000,

64 , ' ,

GIGAPLANE-XB

Sun

Microsystems (16 16). IBM RS/6000 Enterprise Server Model S70

« »

12

RS64.

SMP-

ProLiant 8000

8500

Compaq

Pentium III Xeon

()

CC-NUMA

Ethernet,

RS/6000 SP
Memory Channel, ServerNet

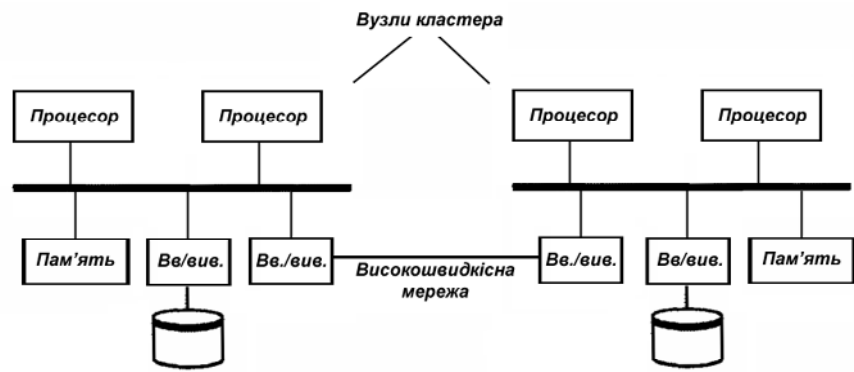
IBM,
Compaq.

Digital

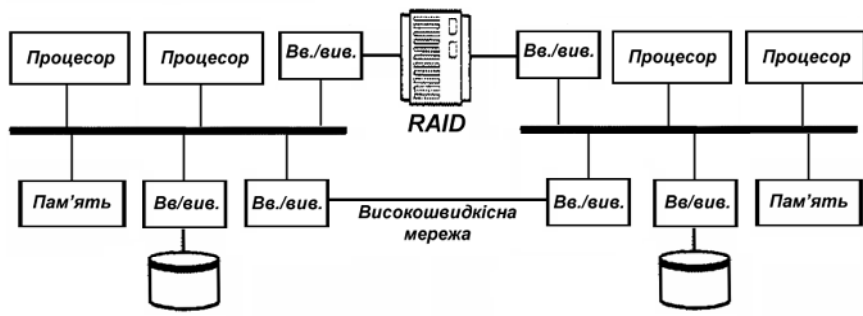
heartbeat,

(),

. 2.7,



а)



б)

. 2.7

. 2.1.

2.1.

()

« »

(. . 2.7,).

RAID(. 2.7,).

()

(Transmission Control Protocol), TCP
- UDP (User Datagram Protocol)

Intel (Microsoft, Compaq .)

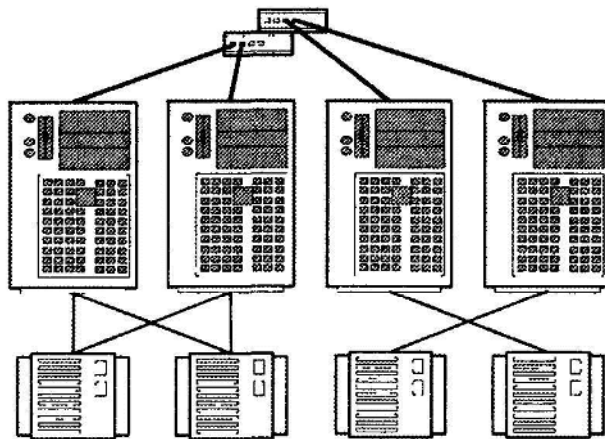
Virtual Interface Architecture (VIA)

« » (keepalive).

()

2-4

(.2.8).



.2.8

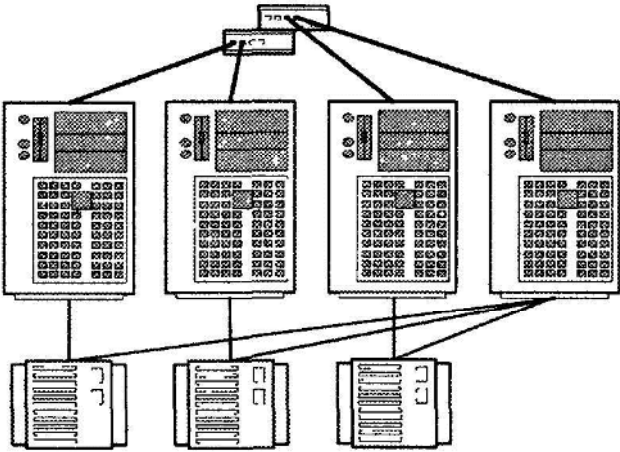
Informix XPS.

$N + 1$

$N+1$

2,3 4

(. 2.9).



. 2.9

$N + 1$

RAID 1 (

3).

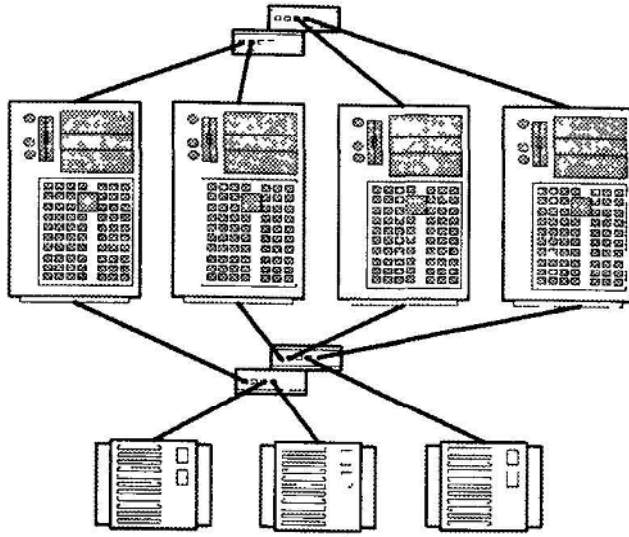
()

$N N$

$N+1,$

$N N (. 2.10)$

2, 3 4 ,



. 2.10 N N

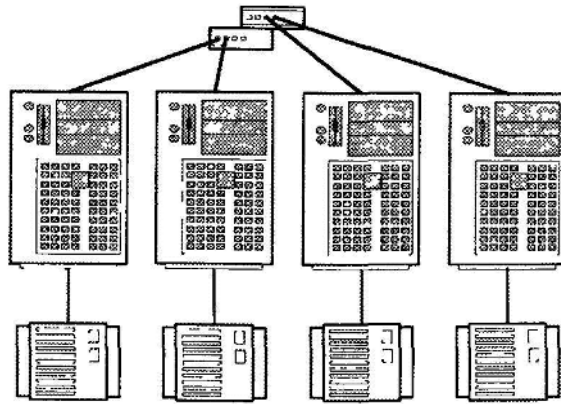
, , , RAID 1 ().

N N

Oracle Parallel Server,

(. 2.11)

Informix XPS.



. 2.11

2.1.4

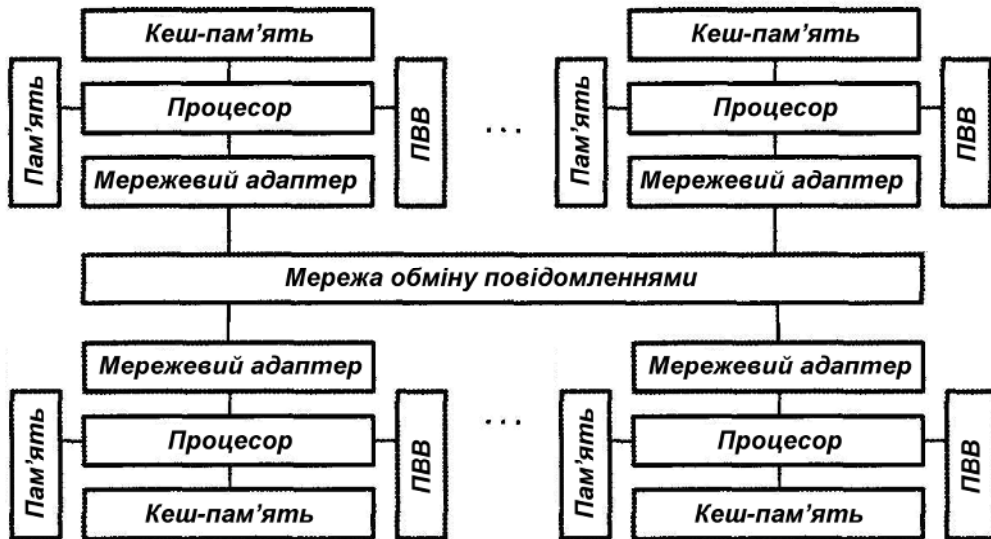
()

Processing),

(, Massively Parallel

> 32

. 2.12.



. 2.12

,
 , :
 • ; ;
 • , ;
 • ;
 • ();
 • MIMD- ;
 • , .
 , - , -
 , ,
 , . - ,
 ,
 MIMD
 , / ,
 ,
 ,
 ,
 (),
 (),
 :
 • ,
 ;
 • ;
 ;

• , ;
 • , ;
 ;
 , (), —
 .
 , .
 , , -
 , ,
 , , SPP1000/XA SPP1200/XA
 , ,
 , - ,
 , .
 , -
 ; -
 , Cray Jaguar, 224162 ,
 6- Opteron, 1,75
 (10^{15}).
 , IBM Roadrunner 12960
 , PowerXCell 8i 3.2
 , 1,04
 .
 , -
 , , .
 , ,

SNI/Pyramid RM1000)

(nCube).

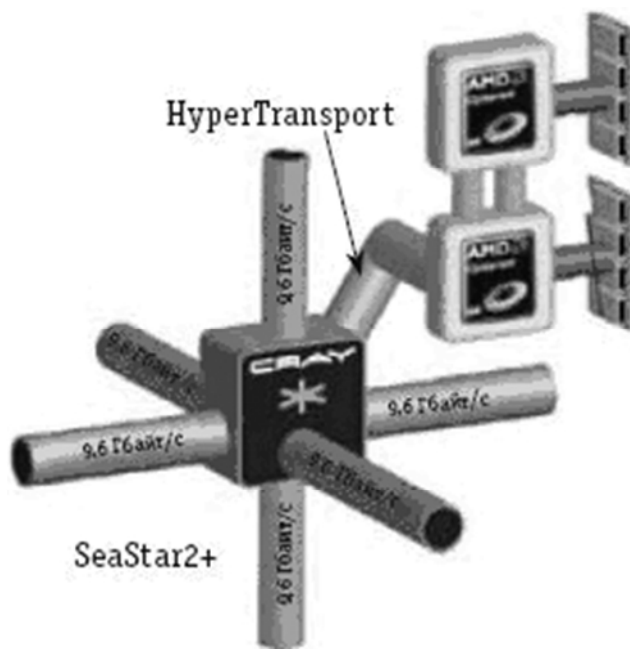
$D.$

$$D \sim 2(m-1), \quad m -$$

$$D \sim n, \quad n -$$

/ . , . , .
 , Cray XT5 Cray XT6,
 , Cray
 Research,
 , Roadrunner, , ,
 Linpack 1 PFlop/s. 31-
 (2008), TOP500 ,
 , IBM
 -
 - , .
 , Roadrunner 6120
 AMD Opteron, 1.8 ,
 12240 IBM Cell 8i, 3.2 ,
 - TriBlades,
 Infiniband. 560
 , 226 .
 - 2.35 , 437 MFlop/s/ .
 IBM Roadrunner 133 .
 , 1.376 PFlop/s, Linpack -
 1.026 PFlop/s.
 - TriBlade Opteron,
 PowerXCell 8i , 16 , Opteron 16
 , Cell. TriBlade LS21,
 QS22. LS21 Opteron
 16 , , 4 . QS22
 PowerXCell 8i 8 , , 4 .

QS22 , PCI x8 LS21,
 QS22. Infiniband 4x DDR.
 - TriBlade , TriBlades
 BladeCenter H.
 49 , Cell 49
 Opteron.
 Roadrunner Red Hat Enterprise Linux
 xCAT.
 RoadRunner
 .
 ,
 , Cray XT53-
 XT6.
 Cray XT3 Opteron ,
 2005 , ,
 . XT4 Opteron 2007 .
 5 Barcelona/2,6 (Shanghai/2,7
) Istanbul/2,6 , 2008 .
 , XT5 .
 Cray Cray XT6 -
 Opteron 6100.
 - Cray XT ,
 , , « » SeaStar
 . Cray XT5
 6. 5 ASIC- ,
 , SeaStar2+ , Opteron
 DIMM (.2.13). Cray XT4
 , .



. 2.13

SeaStar2+

Opteron

HyperTransport 2.0 (

Opteron).

25,6

(

DDR2-800).

8

32

70 GFLOPS

124

GFLOPS.

XT5 – (Blaid),

Istanbul

500 GFLOPS;

1 2 2.

1 4 24 (24

)

12 TFLOPS

1,54

(

16

),

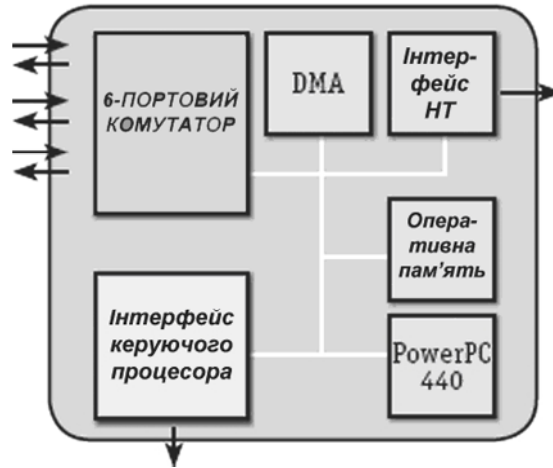
XT5

–

25 32 24.

2 PFLOPS

300
5,
2 , 700
0,6 1,4
43 .



.2.14

SeaStar2+

SeaStar2+

(.2.14),

9,6 / .
57,6 / ,
2 .

HyperTransport, SeaStar2+

(Direct Memory Access, DMA),

SeaStar2+

Cray XT4

XT5, Cray XT5 – XT6 (

), 6

Cray Gemini,

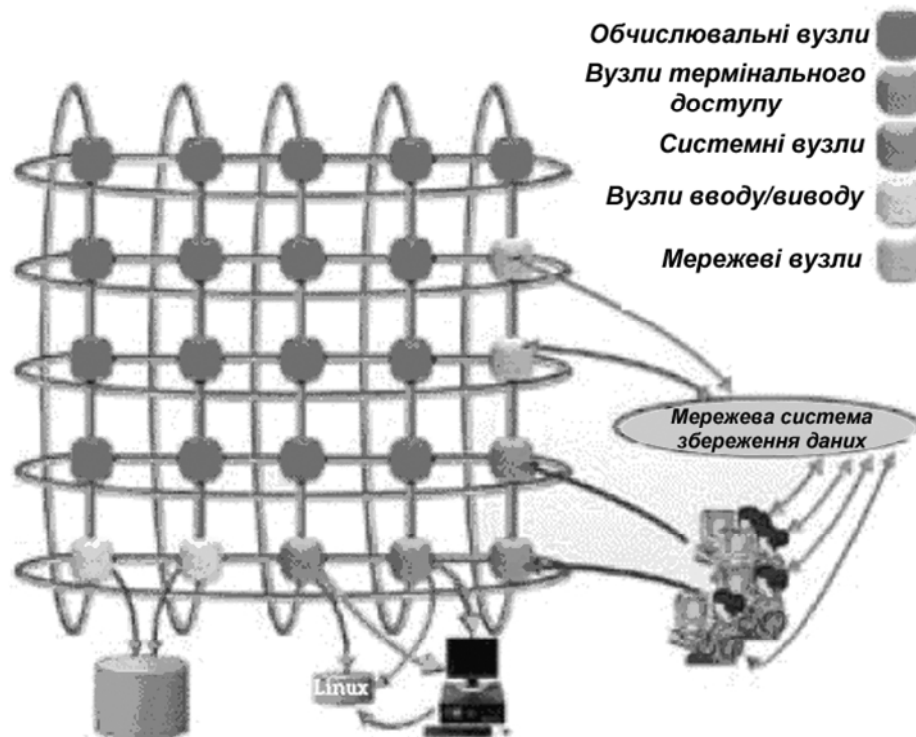
Baker.

AMD Bulldozer 12 16 ,

32-

(2.15), « » PCI-Express. Gigabit Ethernet, 10 Gigabit Ethernet, Infiniband, Fibre Channel. Fibre Channel SATA.

, Cray XT. , Cray SIO- (Storage-I/O).



2.15

Cray XT

. Cray Lustre -

; , 1 ray Sun Microsystems.

2.1.5

,

,

- SMP,

.

,

,

’ .

,

’ (NUMA, Non-Uniform Memory Access),

-

’ (CC-NUMA).

(SMP)

,

.

- ’

’ ,

.

,

.

SMP

16

64.

,

’ SMP-

Silicon Graphics Power Challenge

64

R10000,

SMP

,

,

«

»

,

,

,

,

.

CC-NUM .

, NUMA-

Silicon Graphics Origin

1024

R10000,

Sequent NUMA-Q

,

252

Pentium II.

. 2.16

CC-NUMA.

,

,

, SMP-

(L1)

(L2)

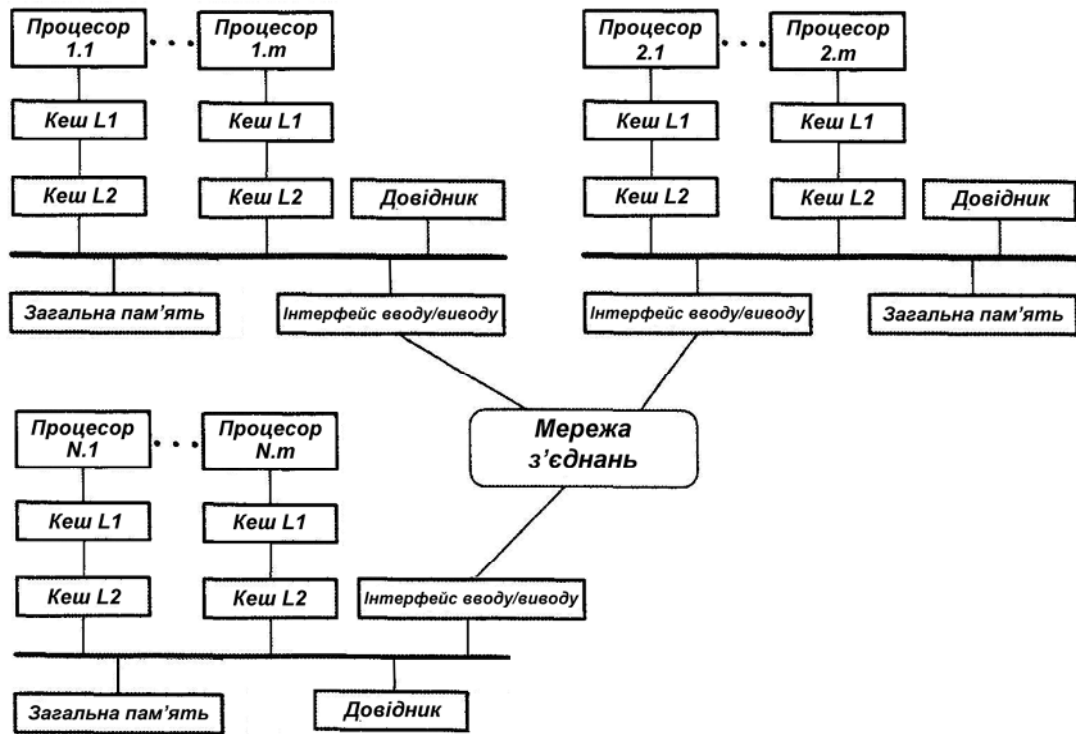
CC-NUMA

Silicon Graphics Origin

MIPS R10000,

Sequent NUMA-Q

Pentium II.



. 2.16.

CC-NUMA

CC-NUMA,

2.2

2.2.1

(point-to-point). N , (x, y)

(source node) s_c , (x, y)

(recipient node) r , (x, y)

(f_c, f_{xy}) - (t_c, t_{xy}) — (l)

b $b_c = w_c f_c$

$(v): l = vt_c$

(direct networks)

(indirect networks)

. (<< >> << >> >> << >>.)



—

,
(),

.

,

,

.

,

.

,

.

.

,

,

—

—

,

,

,

,

.

.

,

.

,

,

.

,

.

,

,

.

.

,

,

«

»

,

(

,

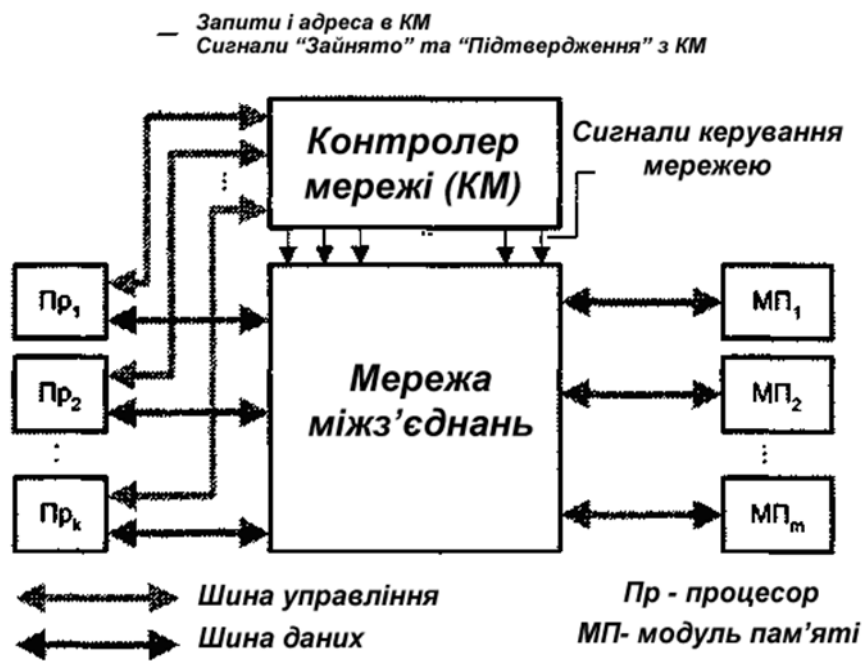
«

» (tail drop),

,

).

(. 2.17).



.2.17

2.2.2

:

- (N);
- (I);
- (D);
- (d);
- (W);
- (T);
- (Q);
- (B);
- (b).

(network size)

(number of links) —

(network diameter),

$$= \{c_1, c_2, \dots, c_n\}$$

(,),

D — (,)

, « »

$$C_{lx} = l_x U_0, \quad l_x = \{ : r = \} —$$

, $aC_{0x} = \{c:s_c = x\} —$

$$| C_{lx} / \quad | \quad 0x /$$

(, ,),

4096 (2¹² 64²),

126, — 12.

(network bandwidth)

(network latency) — ,

.

,

.

(network connectivity)

,

.

,

.

(bisection width).

(cut of network). $C(N_1, N_2)$ — ,

N

N_1

N_2 .

(N_1, N_2) — ,

N_1 ,

N_2 .

—

,

, $|N_2|$ $|N_1|$ $|N_2| + 1$.

,

:

$$B = \min_{bisection} |C(N_1, N_2)|. \tag{2.1}$$

,

,

,

.

(bisection bandwidth) —

.

,

,

,

,

.

b

$$b = \min B(N_1, N_2).$$

b

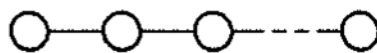
$$: b = b_c \times .$$

2.2.3

- () ;
- (, , , ,) ;
- (' , ,) ;

(. 2.18).

: $D=N-1$; $d=2$; $I=N-1$; $=1$.



. 2.18.

1,

2.

(2.19,).

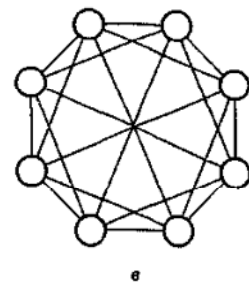
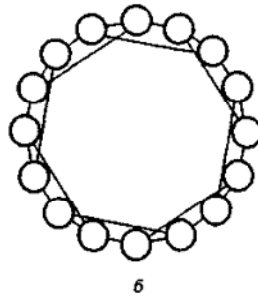
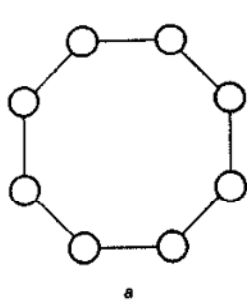
()

$$D = \min \left[\frac{N}{2} \right]; d = 2; I = N; B = 2.$$

1 (N/2)-1,

2.19,

3.



.2.19.

: — ; — ;

Ring,

IBM,

Token
KSR 1 S I.

. 2.3,

N

$i \ j,$

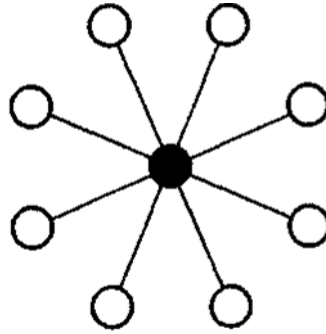
$|i - j|$

2.

2.20).

1; $B=1$.

— (.
: $D = 2$; $d = N - 1$; $I = N -$

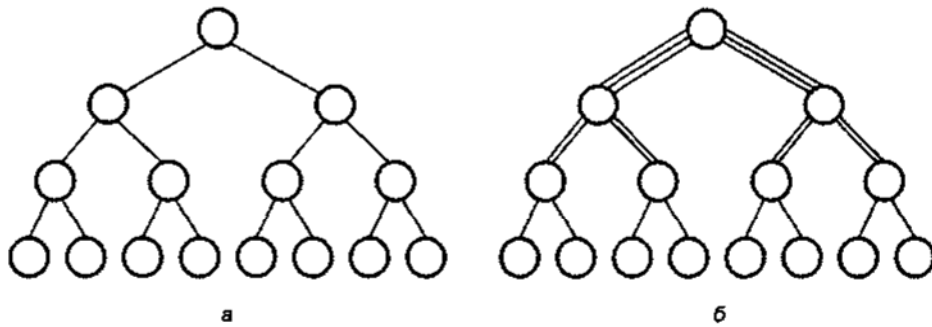


. 2.20

2.21,).

h — (),
 $\max[\log_2 N]$,
 $: D = 2(h-1); d = 3; I = N-1; B=1.$,
 262 144 ()
 512, — 36.

DADO 1023 , .



.2.21.

— ; — « » :

« » (.2.21,).

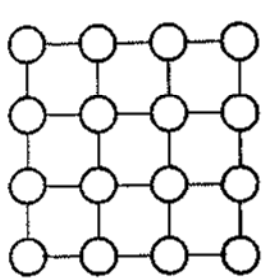
« »

, ,

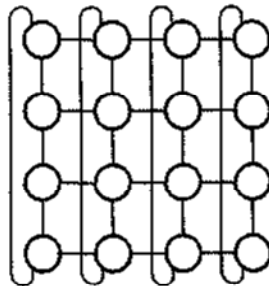
« »

-5.

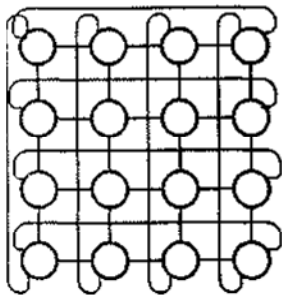
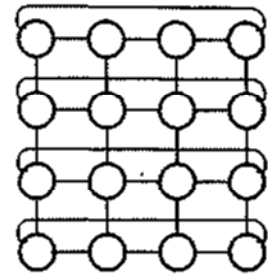
(mesh),



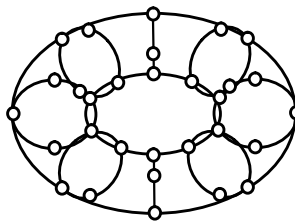
a



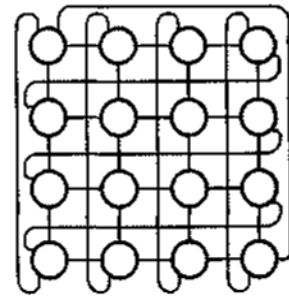
b



d



e



f

. 2.22.

- —

: —
; —

; —

;

(. 2.22,).

$$: D = 2(- 1); d = 4; I = 2N-2m; B = .$$

(wraparound)

(. 2.22,).

(. 2.22,).

$$D = 2 \min \left[\frac{m}{2} \right]; d = 4; I = 2N; B = 2m. \tag{2.2}$$

4x8

. 2.22, .

(twisting).

ILLIAC. . 2.22,

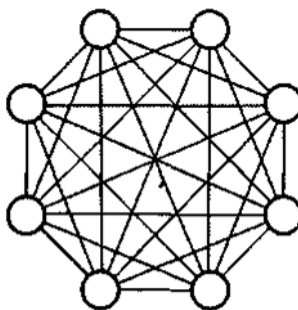
$$: D = m - 1; d = 4; I = 2N; = 2 .$$

Cray T3D,

: ILLIAC IV, , DAP, CM-2, Paragon .

,
 , (. 2.23),
 « »
 . , N ,
 :

$$D=1; d=N-1; I=\frac{N(N-1)}{2}; B=\frac{N^2}{4}. \quad (2.3)$$



.2.23 ,

, ,
 . , ,
 , ,
 N - 1 .
 , ,

, .2.24. , (.
 2.24,), . ,
 (.2.24,), — , 8 (.2.24,)

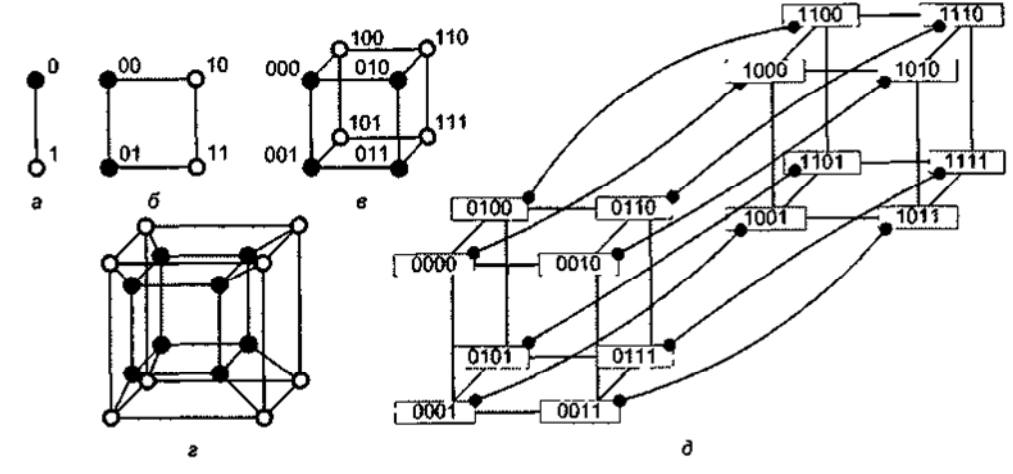
— .
 m- : (-I) - ,
 ,

$(N = 2^m)$

$$D = m; d = m; I = \frac{mN}{2}; B = m. \tag{2.4}$$

1

0010
 0110 — , 0110 0101 — .
 m- (- 1) - .2.8, .
 (- 1) - (-
 1) -



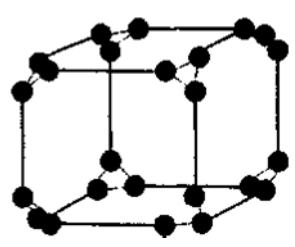
.2.24.

— ; — ; — ; — ; — ; — ; —

m-

$i-0$, $i-0$, $(i-1)-$.

(2.25).



2.25

$k-$ $n-$

k ($N = k^n$).

$n-$ $k,$

$k-$

$n-$ k $k-$ (

$-1)-$.

$k-$ $n-$:

$k-$ 1- — ;

$k-$ 2- — ;

$k-$ 3- — ;

4- 2- — 4x4;
 2- n- — .
 , ,
 k .

. 2.2.

2.2.

	1	N-1	$\frac{N(N-1)}{2}$	$\frac{N^2}{4}$		N
	2	1	N-1	1		N
	2(h-1)	3	N-1	1		h = log ₂ N
	N-1	2	N - 1	1		N
	$\frac{N}{2}$	2	N	2		N
	2(m-1)	4	2N-2m	\sqrt{N}		m =
	2[m/2]	4	2N	2		m = \sqrt{N}
			$\frac{nN}{2}$	$\frac{N}{2}$		N = log ₂ N ;
n-	n[k/2]	2n	nN	2k ⁿ⁻¹		N = k ⁿ

2.2.4

(),

,

(two-sided).

(one-sided).

.

-

-

(— permutations).

(multicast),

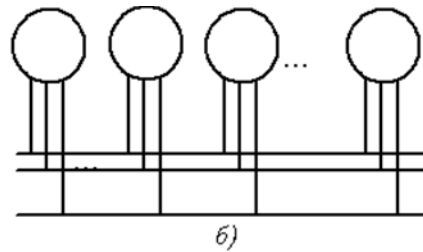
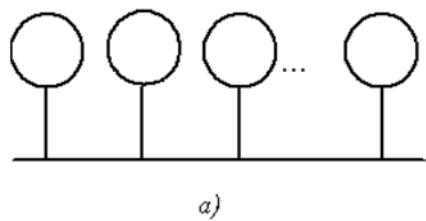
()

n

:

(strictly non-blocking)
blocking).

(wide sense non-



. 2.26

: -

; -

(« »)

(crossbar

switch system)

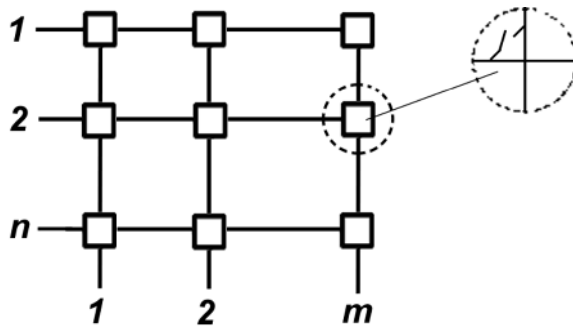
()

« »,

()

(. 2.27) -

$\min(,)$.



.2.27

$n \ m$

« \dots » = , « \dots » .
 n^2 .
 1, — $n/2$.
 () , .

256 .

VPP500 224 224.

Fujitsu

• ;
 • .
 « \dots » 2×2 .

(. 2.28).

() -

:

.

,

.

.

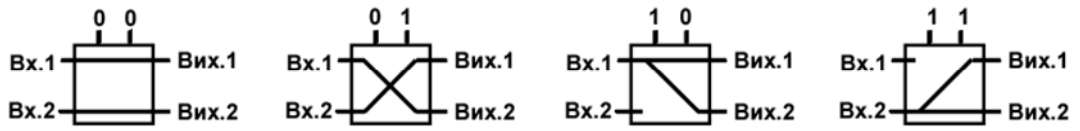
.

-

,

,

.



. 2.28.

-

-

. 2.29.

()

-

,

,

.

-

.

.

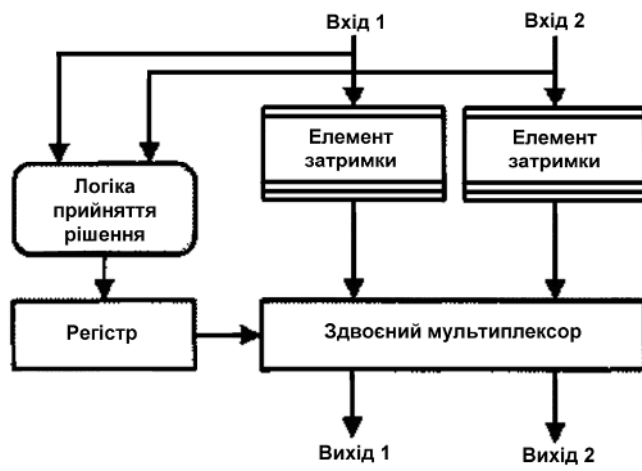
,

,

-

,

.



. 2.29

2.2.5

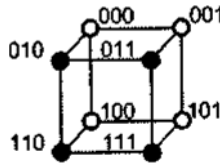
N , \dots , $= \log_2 N$.
 b_i .

(exchange)

:

$$E_k(b_m, \dots, b_k, \dots, b_1) = (b_m, \dots, \bar{b}_k, \dots, b_1), \quad 1 \leq k \leq m. \quad (2.5)$$

(. 2.30).



. 2.30

(shuffle)

:

(perfect shuffle),

(unshuffle),

i -

(i^{th} subshuffle)

i -

(i^{th} supershuffle).

, . 2.31 —

I

:

$$S(b_m, b_{m-1}, \dots, b_1) = (b_{m-1}, b_{m-2}, \dots, b_1, b_m). \quad (2.6)$$

i

$i \ j$

j

—

:

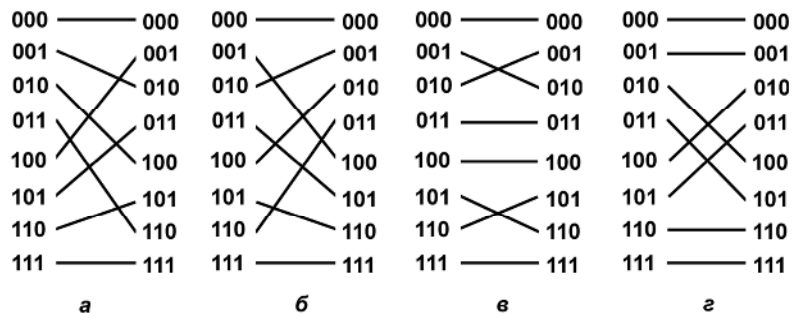
$$U(b_m, b_{m-1}, \dots, b_1) = (b_1, b_m, \dots, b_2). \quad (2.7)$$

i - :

$$S_i(b_m, b_{m-1}, \dots, b_i, \dots, b_1) = (b_m, \dots, b_{i+1}, b_{i-1}, \dots, b_1, b_i). \quad (2.8)$$

i - :

$$S^i(b_m, b_{m-1}, \dots, b_i, \dots, b_1) = (b_i, b_{m-1}, \dots, b_{i+1}, b_m, b_{i-1}, \dots, b_1). \quad (2.9)$$



. 2.31.

$m = 3$:

— ; —
— ; —

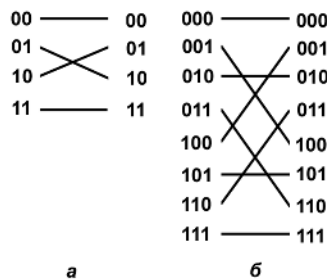
« » (butterfly) — « »

60-

(. 2.32).

:

$$B(b_m, b_{m-1}, \dots, b_1) = (b_1, b_{m-1}, \dots, b_2, b_m). \quad (2.9)$$



. 2.32

« » :

- $m = 2$; - $m = 3$.

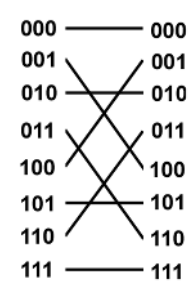
« »
 ,
 « » « »- .
 ,
 :

$$R(b_m, b_{m-1}, \dots, b_1) = (b_1, b_2, \dots, b_m). \tag{2.9}$$

= 3 . 2.33.

= 3

« », .



. 2.33

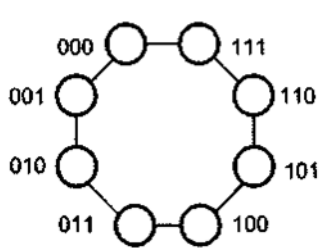
(m = 3)

:

$$SX(x) = (x+1) \text{ mod } N \quad (N = 2^m) \tag{2.9}$$

= 3

(. 2.34).



. 2.34

(m = 3)

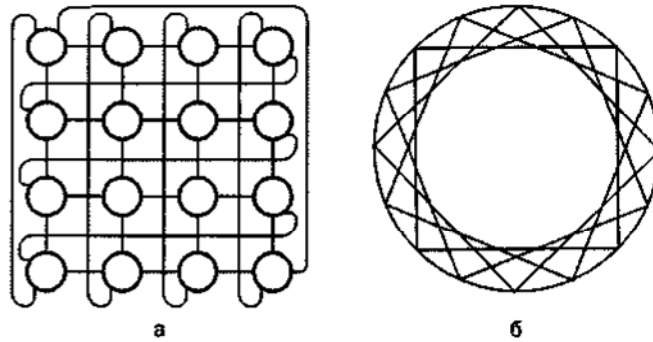
ILLIAC IV

« »

ILLIAC IV,

ILLIAC IV:

$$\begin{aligned}
 R_{+1} &= (i+1) \bmod N; \\
 R_{-1} &= (i-1) \bmod N; \\
 R_{+r} &= (i+r) \bmod N \quad (0 \leq i \leq N-1); \\
 R_{-r} &= (i-r) \bmod N \quad (r = \sqrt{N}).
 \end{aligned}
 \tag{2.10}$$



. 2.35

ILLIAC IV:

$$N=4$$

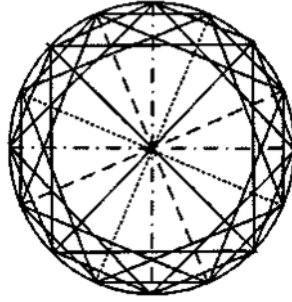
. 2.35.

4.

(barrel shift)

$$\begin{aligned}
 B_{+1}(j) &= (j+2^i) \bmod N, \\
 B_{-1}(j) &= (j-2^i) \bmod N, \quad 0 \leq j \leq N-1, \quad 0 \leq i \leq \log_2 N-1.
 \end{aligned}
 \tag{2.11}$$

. 2.36.



. 2.36

$\pm 2i$

Serial Attached SCSI (SAS),

3 / ,

10

/ .

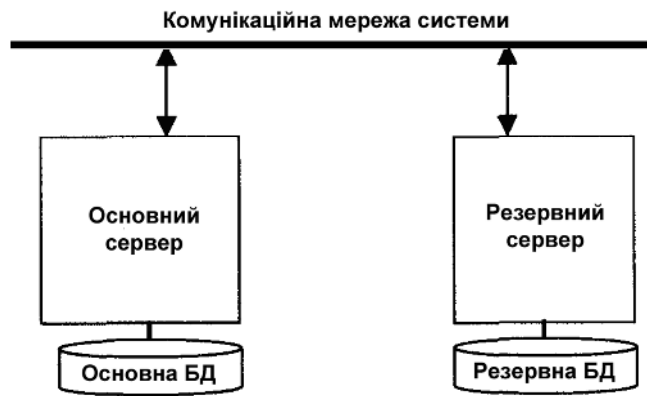
. 2.37.



. 2.37

()

. 2.38.



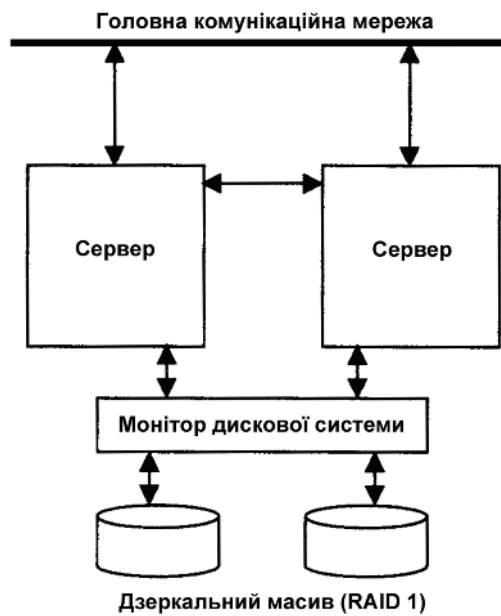
. 2.38

. 2.39.

Teradata, Oracle Informix,

nothing.

sharing



.2.39

2.3.2

(HDD).

RAID.

«RAID» (Redundant Arrays of Inexpensive

Disks)

«

».

RAID

1987

RAID –

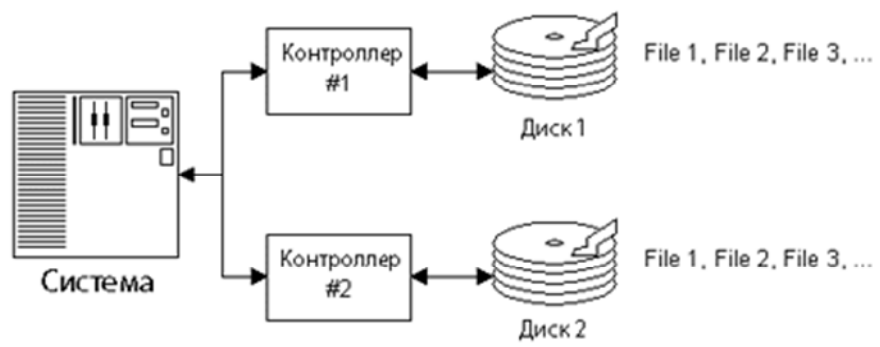
RAID-

RAID
IDE/ATA SATA

RAID (Array),
(Mirroring), (Duplexing), (Striping)
(Parity).

RAID

RAID (.2.40).



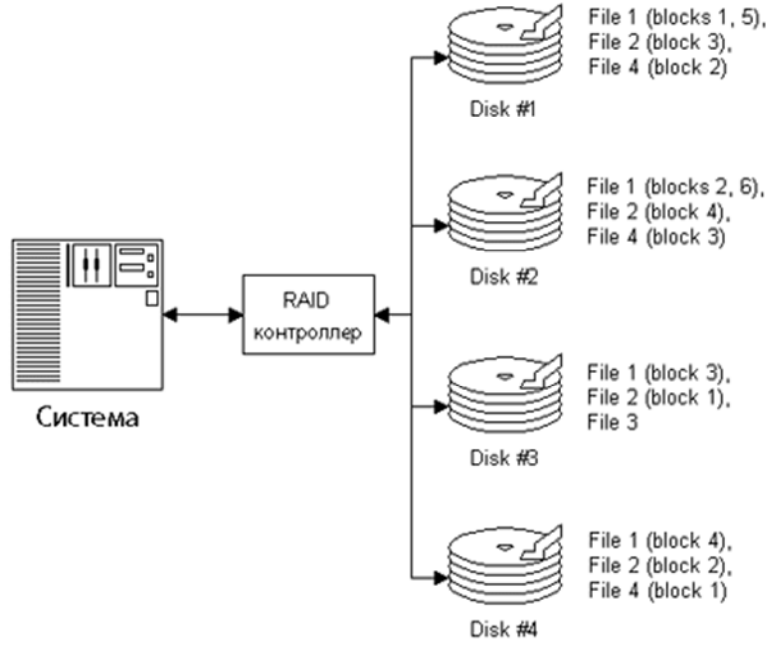
.2.40

RAID-

(.2.41) .

« » .
« » – 1 ,
– 512 ().

() () .



. 2.41

N

, $N+1$

RAID-

$N+1$

?

N

XOR.

XOR

$$(A \text{ XOR } B) \text{ XOR } B = A.$$

5

20%

RAID-

, RAID

(single)

(multiple)

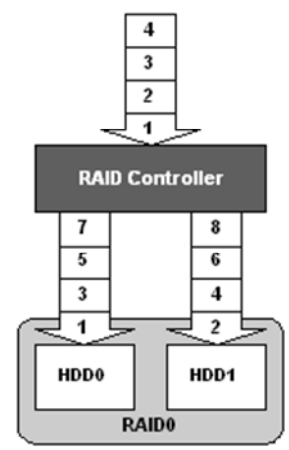
RAID.

8-

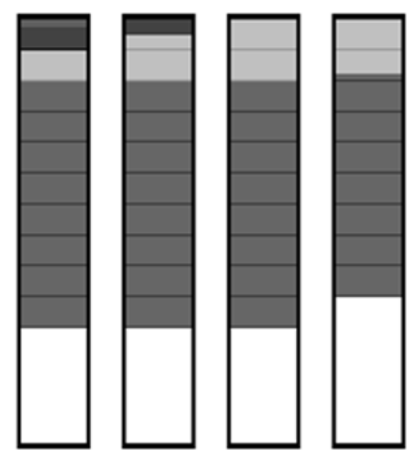
RAID 0

(.242) .

(, 16)



a)



b)

. 2.42 RAID 0. a) -

,) -

RAID 0 4
 - 16
 - 20
 - 100
 - 500

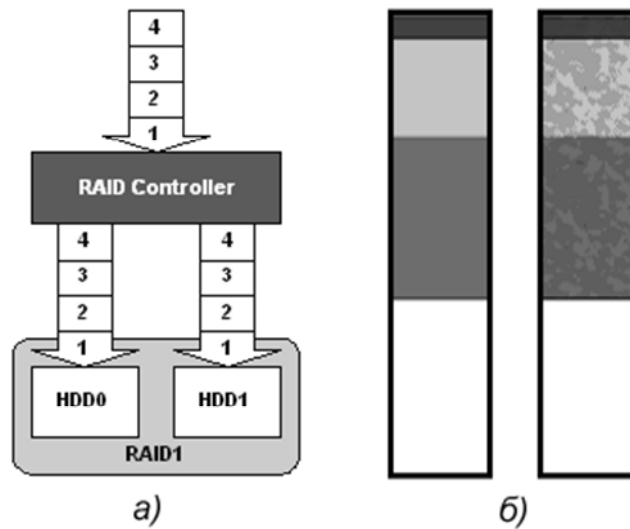
RAID 0

RAID-

RAID 1

(.2.43) .

RAID.



. 2.43 RAID 1: a) -

,) -

RAID 1

RAID 1

RAID 2

RAID

(.2.44)

14

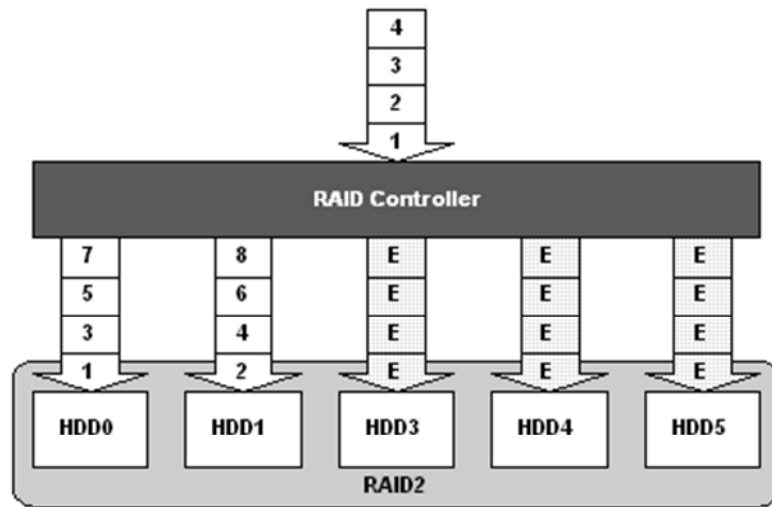
39

(!).

(10

32)

RAID 2



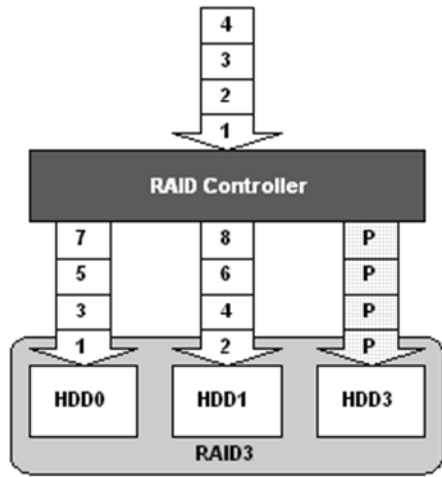
.2.44

RAID 2

RAID 3

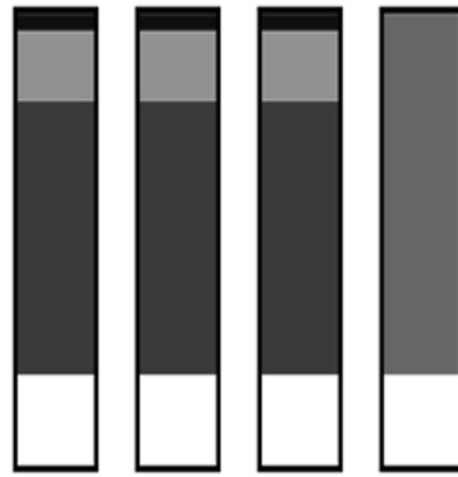
(.3.16)

1024



a)

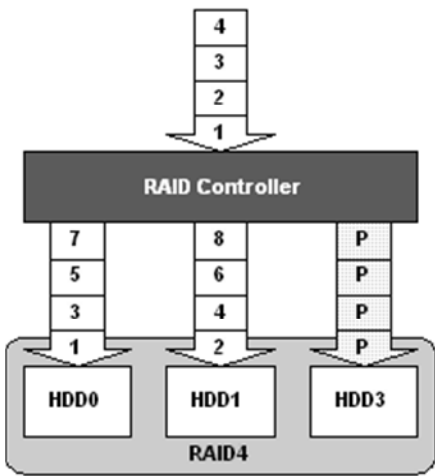
. 2.45 RAID 3: a) -



b)

,) -

RAID 4



a)

. 2.46 RAID 4: a) -



b)

,) -

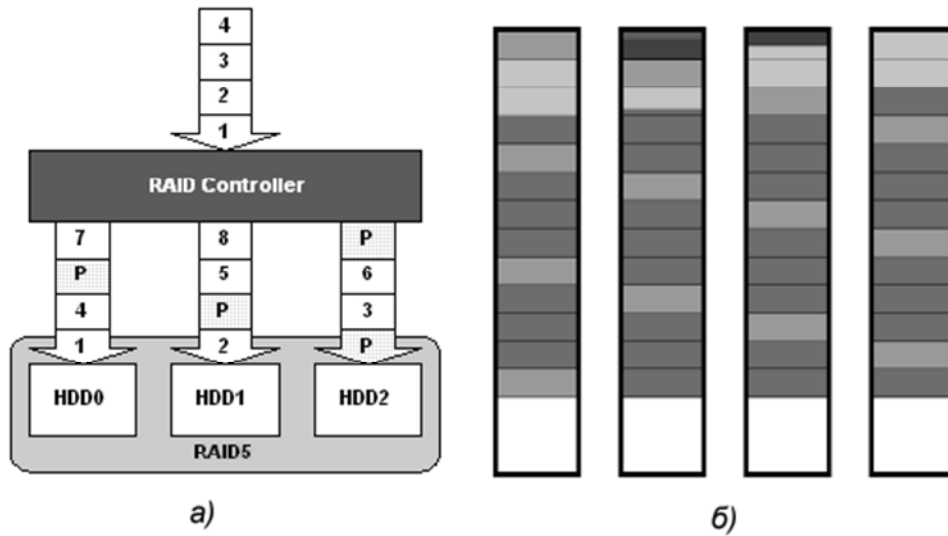
RAID 3

(.2.46) .

« »

RAID 3 RAID 5,

RAID 5



.247 RAID 5: a) -

,) -

RAID 3,

— ,
(.247) .

RAID

RAID 6

. , . ? , .

RAID 5

, RAID 6 , - , .

RAID 7

, RAID 7 ,

RAID 3

Storage Computer Corporation.

, - ,

, RAID 3, , RAID 7

RAID

RAID

RAID,

RAID -

RAID 0

RAID 1, 3 5.

RAID

(set).

X+Y

RAID X,

RAID X

RAID Y.

RAID 0+1 (01) 1+0 (10)

RAID 0+1

«

»

RAID 1+0 –

«

» (

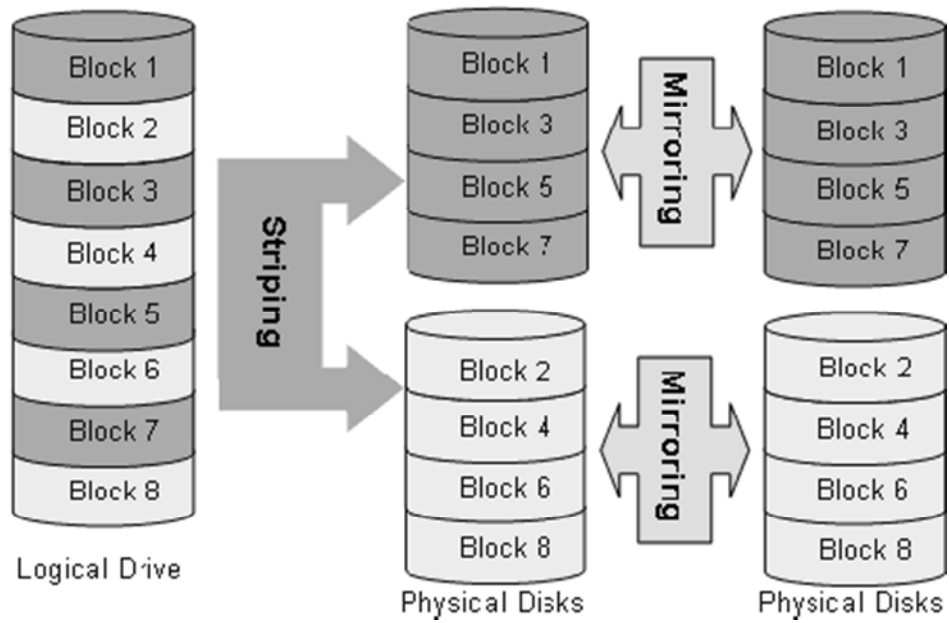
«

»

).

–

(.248),



. 2.48

RAID 0+1

RAID 0+1

RAID

RAID 1+0.

10

(5 2)

5

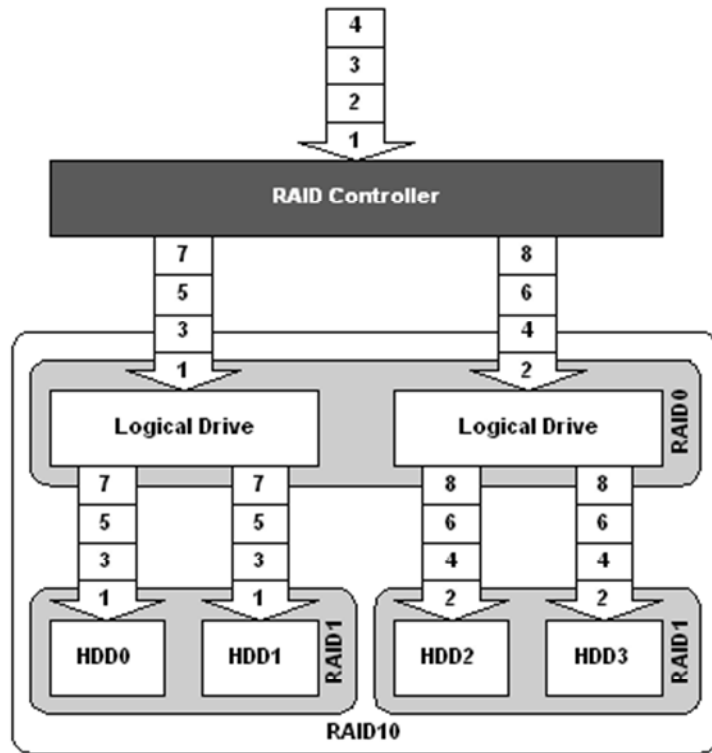
!

10

50%.

RAID 01

(.2.49) .



. 2.49

RAID 1+0

RAID 0+3 (03) 3+0

0+3 3+0

5+3 (53).

!

RAID 3

3+0.

RAID 3

- 6. RAID 3+0

0+3.

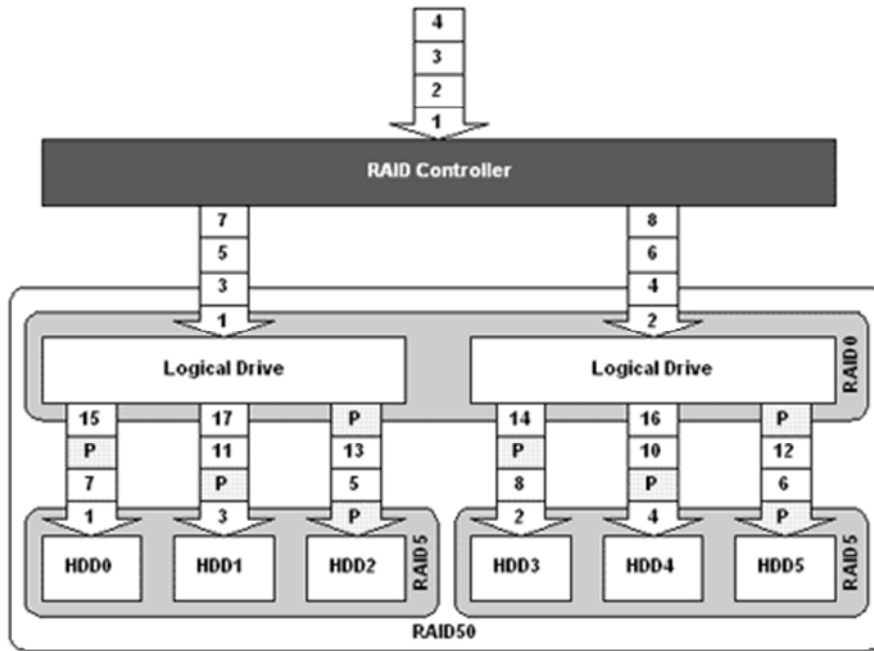
RAID 0+5 (05) 5+0 (50).

?

. RAID 0+5

RAID 5.

RAID 5+0 (.2.50).



. 2.50

RAID 5+0

RAID 5,

WEB-

RAID 1+5 (15) 5+1 (51)

RAID 15 51 -

1+5

, 5+1 -

RAID 5+1

RAID 5,

RAID

2.3

RAID-

2.4

RAID-

2.3 -

RAID-

	RAID 0	RAID 1	RAID 3	RAID 5	RAID 6
Технологія	Чергування	Дзеркалювання	Чергування, парність	Чергування, парність	Чергування, парність
Контроллер	Програмний	Програмний	Апаратний	Апаратний Ні-End	Спеціалізований
К-ть жорстких дисків	2, 4	2	3 і більше	3 і більше	3 і більше
Доступний робочий простір %	100	50	66 для 3, 75 для 4	66 для 3, 75 для 4	33 для 3 50 для 4 60 для 5
Стійкість при відмові диска	Немає	Висока	Висока	Висока	Дуже висока
Відновлення даних	Немає	Дуже швидке	Швидке	Швидке	Дуже швидке
Швидкість випадкового читання	Дуже хороша	Хороша	Хороша	Дуже хороша	Дуже хороша
Швидкість випадкового запису	Дуже хороша	Хороша	Погана	Нормальна	Погана
Швидкість лінійного читання	Дуже хороша	Хороша	Дуже хороша	Дуже хороша	Хороша
Швидкість лінійного запису	Дуже хороша	Хороша	Хороша	Хороша	Середня
Ціна	Найнижча	Низька	Середня	Середня	Висока

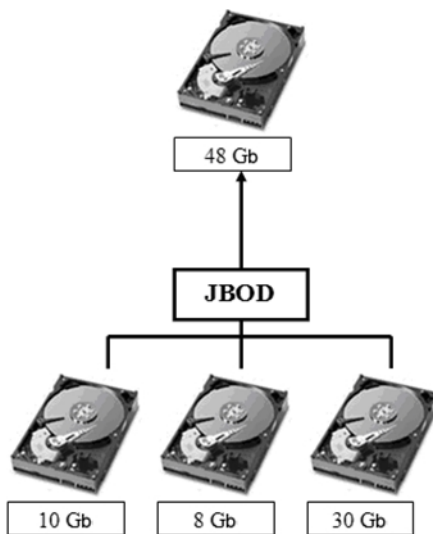
JBOD

?

?

RAID, JBOD – Just A Bunch Of Disks.

(2.51).



. 2.51 JBOD

JBOD

?

2.4 -

RAID-

	RAID 0+1	RAID 1+0	RAID 5+0	RAID 5+1
Технологія	Чергування, дзеркалювання	Чергування, дзеркалювання	Чергування, парність	Чергування, парність, дзеркалювання
Контроллер	Програмний	Майже все	Спеціалізований	Спеціалізований
К-ть жорстких дисків	4 min	4 min	6 min	6 min
Доступний робочий простір %	50	50	66 для 2 страйпів по 3 диски	33-40
Стійкість при відмові диска	Дуже хороша	Відмінна	Хороша	Відмінна
Відновлення даних	Швидке	Дуже швидке	Середнє	Швидке
Швидкість випадкового читання	Дуже хороша	Дуже хороша	Дуже хороша	Дуже хороша
Швидкість випадкового запису	Хороша	Хороша	Хороша	Хороша
Швидкість лінійного читання	Дуже хороша	Дуже хороша	Дуже хороша	Дуже хороша
Швидкість лінійного запису	Хороша	Хороша	Хороша	Хороша
Ціна	Відносно висока	Відносно висока	Висока	Дуже висока

1. , IMD. SMP- ,
2. , SMP- .
3. , .
4. .
5. , .
6. , , .
7. , .
8. , .
9. . , , .
10. ILLIAC. , ,
11. : , , .
12. : , , .
13. : , .
14. , .
15. , .
16. RAID0. RAID1, RAID2, RAID3 .
17. RAID4, RAID5. .
18. RAID6, RAID7. .
19. RAID0+1 RAID3+0.
20. RAID0+1 RAID3+0.
21. RAID5+0 RAID5+1. JBOD.

3. ,
(,).

4. .

5. .

6. .

:

1. .

2.

.

3. .

4. .

5. , (HDD)

.

6.

().

,

.

-

.

-

.

:

-

-

, ();

-

(-

);

-

().

IBM PC,

.

:

.

()

BIOS.

- ;
- () ;
- ();
- ;
- ();
- ();
- () .

3.1.2

.
, :
-
, ;
- , , ;
- , , ;
, , - , . . ,
;
- -
, , , .
.
« »,
,
, , , .
,
,
,
, .
, Windows, Unix.
,
, , , ,
- , , ,
.
, fly-by-wire (- «
» ,
,) ,

MS Windows 3.x Windows 95,

RT-11,

« »

VMWare

i 86

VM

System/370

« » (- « »)

(« ») -

• ;
• () , ' () :
• ;
• ;
• ;
• - ;
• ;
• ;
• . ,
()
(), ,
, - (-)
, ()
, - .
:

1. ()

(

, 4).

2. , .

, (

),

(,

)

3. .

, -

4. ,

2

,

5. . 3,

6. ,

(, 4).

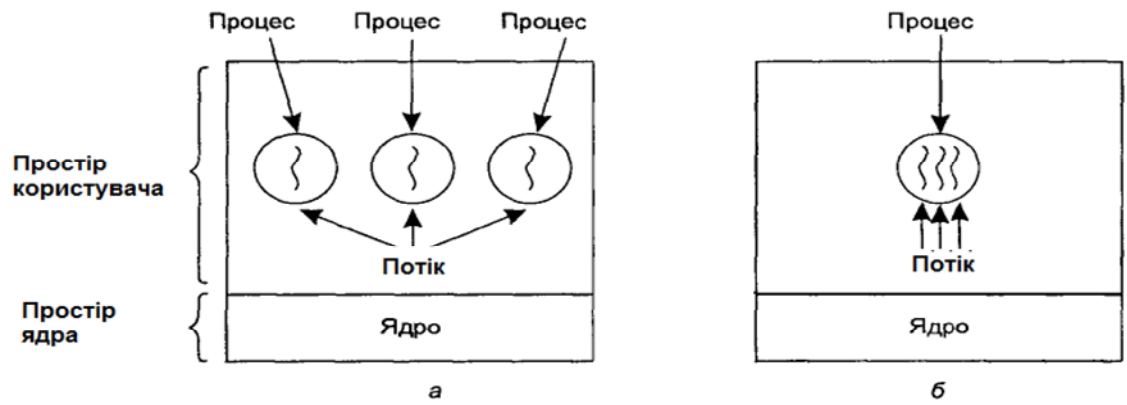
7. .

1-3

, 4-7 - .

3.2.

3.2.1



.3.1.)

)

(

). ,

,

:

'

(shared memory)

,

(

).

,

,

,

,

,

(message passing)

(, ,

)

,

(messages).

,

,

—

,

’ (mapped memory). ’

’ (memory-mapped files).

3.2.2

```

    .
        '
            ,
                ,
                    '
            .
                ,
                    ,
                        '
                            :
                                - - , ' (
                                    , ' ) ;
                                        , ' ,
                                            - - , ' ,
                                                (
                                                    , ' ,
                                                        ) ;
                                                            - - ,
                                                                ,
                                                                    .
                                                                        ,
                                                                            ,
                                                                                ,
                                                                                    .
                                                                                        ,
                                                                                            '
                                                                                                ,
                                                                                                    ,
                                                                                                        .
                                                                                                            ,
                                                                                                                '
                                                                                                                    (
                                                                                                                        ,
                                                                                                                            '
                                                                                                                                )
                                                                                                                                    .

```

Windows Linux.

3.2.3

(communication channel).

(,); ;
, (,).

— ,
: **send** (
) **receive** ().

, send receive

, (,);

send, ?

•
receive, ?

—

send

();

),

(
)

),

send

;

3.2.4

(named pipes)

(message queues).

(sockets).

—

;

TCP/IP) (UNIX,

).

UNIX

UNIX

:

- (stream sockets) —

);

(datagram sockets) —
(
,).

, () ,
, - (,
socket ().

1. , bind(
)). UNIX ,

2. listen() ,

3. listen()

accept().

accept()

connect().

bind().

send(),

— recv().

(Remote Procedure Call, RPC)

1. send

2.

RPC

3. receive

4.

5.

3.3.

3.3.1

— , , ,
,
.
—
(, ,),
,
,
,
:
• ;
• , ;
• , ;
• , .
.
, :
, :
1. , , , ,
, , , , ,
- , ()
, ,
,
1 3, 10 5 10⁸ .
, 200
, 10¹¹
10 10- ,
100 Mflops (10⁸)

10⁷ 100 . ,
 10 , 1.7 Tflops (1.7X10¹²)
);

2. , , , ,
 , , -
 -

· , , , ,
 , 10 .

N⁴ N⁵, N .
 ;

3. , , ,
 - ;

4. , , , ,
 ;

- :
 • () ;
 • , ;
 • , ;
 • ;
 • ;

- :

- (: , , , . .);
- : (PVM, MPI); (Dynamite, Globus).

3.3.2

, .3.1. " "

, , , (,).

, .

3.1.

		,	
		/	
			MIMD-
			SIMD-
	-		

pi .

i i

.

(,)

(,)

).

(. ,)

, (, ,) .

(Scheduler)

.

, .
(), , , ,

" "

pi

" " i

, , .

pi

() -

i

,

-

,

.

,

, ,

,

n*n

n*n

(

).

,

,

,

.

-

.

,

,

“

.

”.

pi

.

-

.

,

8-

-

3.3.2

.

,

, i

,

.

,

,

,

.

,

i

.

,

.

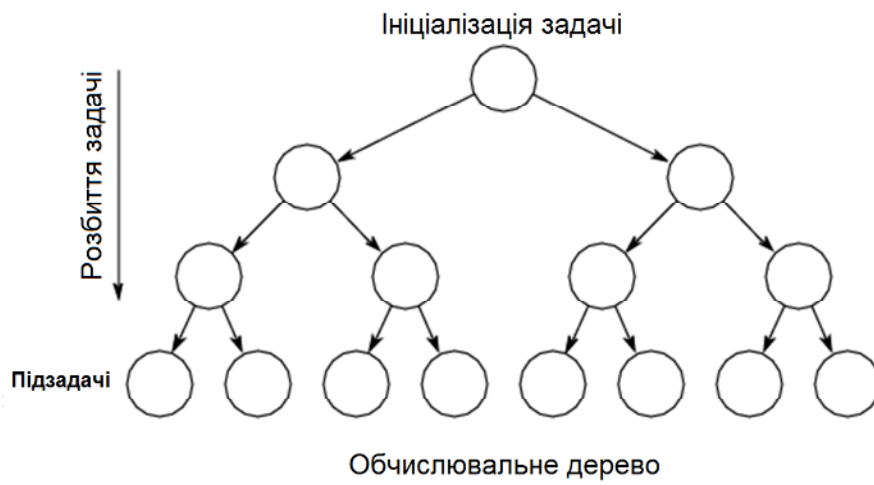
,

cix

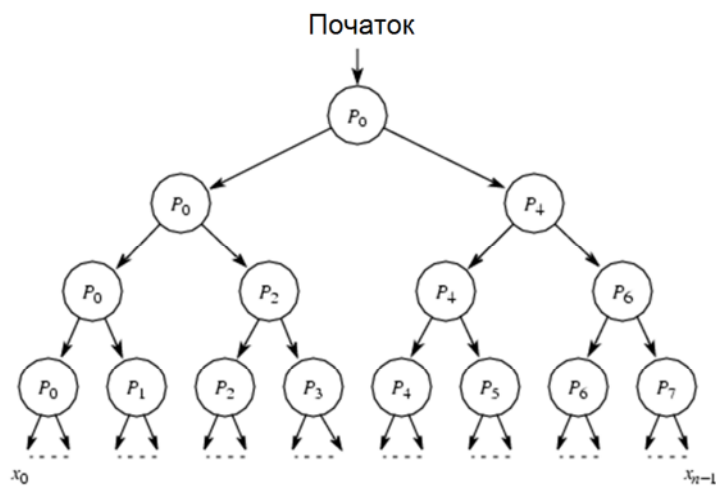
,

.

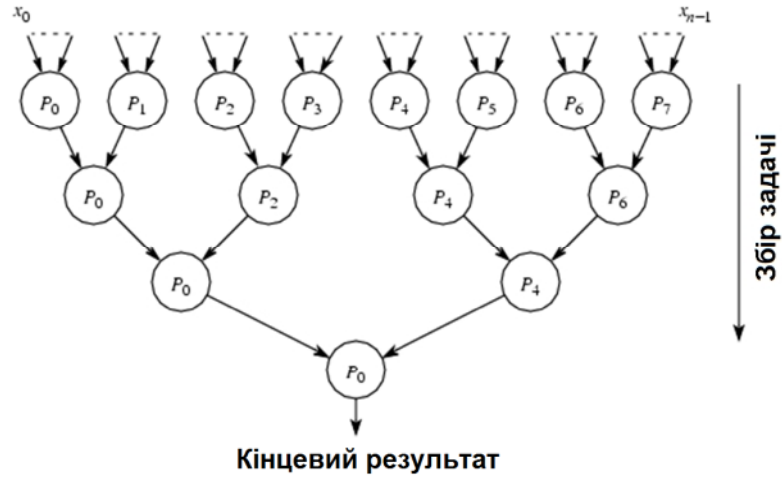
(3.2-3.4).



. 3.2



. 3.3



. 3.4

3

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.
- 8.
- 9.
- 10.
- 11.
- 12.
- 13.
- 14.

?

?

?

?

?

?

?

4.

, ' -
,

4.1.

- ()

4.1.1

, -

,

,

,

,

,

..

-

.

,

,

,

.

-

,

,

;

-

.

,

.

,

,

.

()

,

(

())

.

,

,

,

,

,

).

.

(-)

,

.

-

-

.

;

.

,

,

,

.

,

.

,

,

.

()

.

.

,

,

,

,

HDD,

.

;

,

-

,

,

-

.

4.1.2

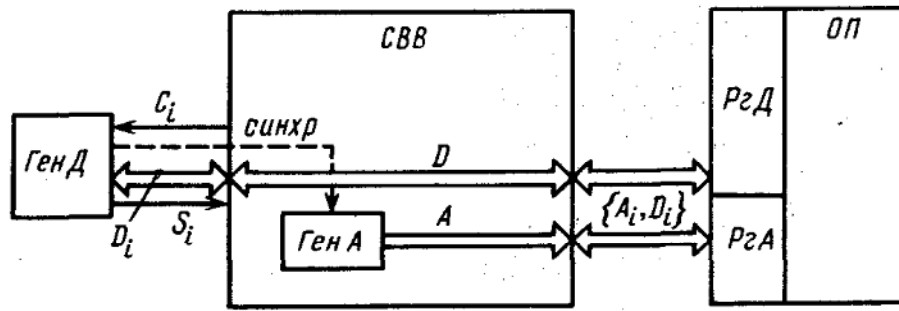
()

()

S_{ij}

D_{ij}

4.1.



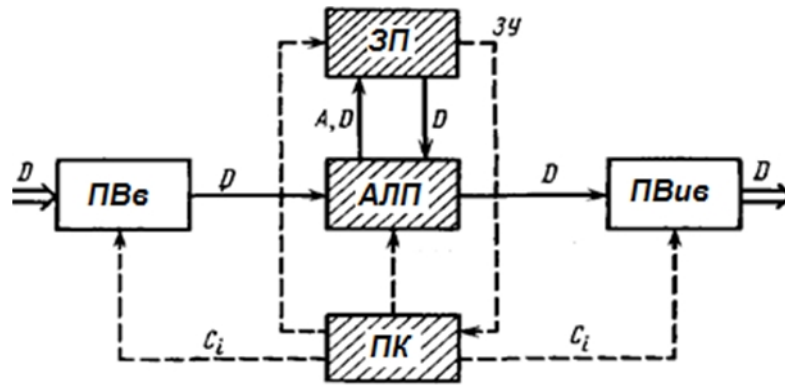
4.1

ij, с
 D_{ij} .

S_{ij} ,

(),

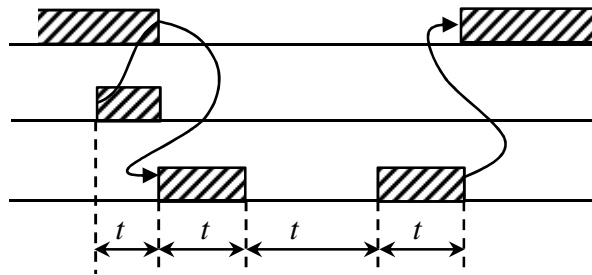
. 4.2.



4.2

()

4.1.3



.4.4

().

4.2.1

4.2.

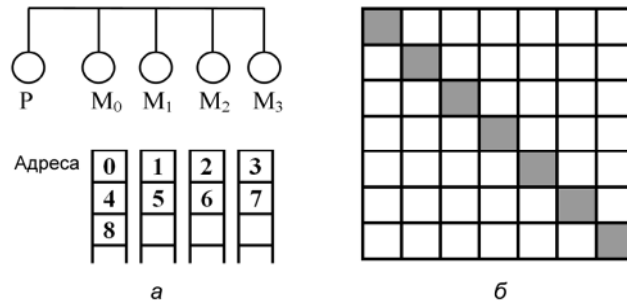
() ,
(shared memory)

(distributed memory).

(closely coupled systems).
 SIMD, MIMD.
 SM-SIMD (Shared Memory SIMD) SM-MIMD
 (Shared Memory MIMD).
 (loosely
 coupled systems).
 SIMD, MIMD,
 DM-SIMD (Distributed Memory SIMD) DM-MIMD
 (Distributed Memory MIMD).
 load R0, «
 R0 ».
 R0, — load
 R0

4, , v_1, v_4, v_8 .
 « » (stride).

(. 4.5,).



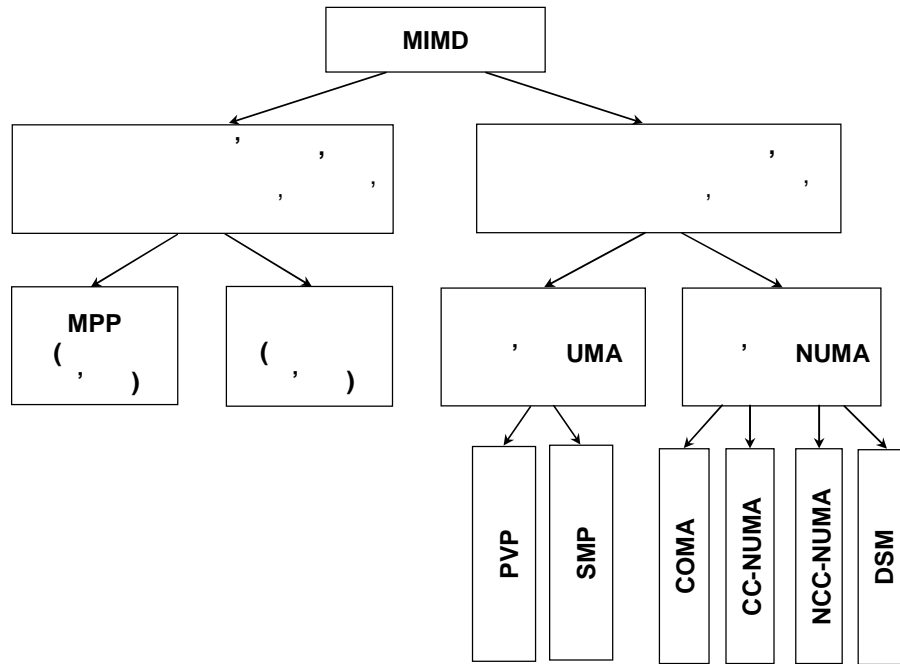
. 4.5 , : — 9 8 x 8.

4.2.2

. 4.6

MIMD (SIMD).

UMA (Uniform Memory Access).



. 4.6.

UMA-

— , (i) , ()

. 4.7, .

j (i j)

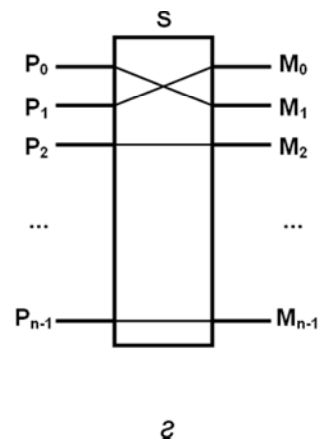
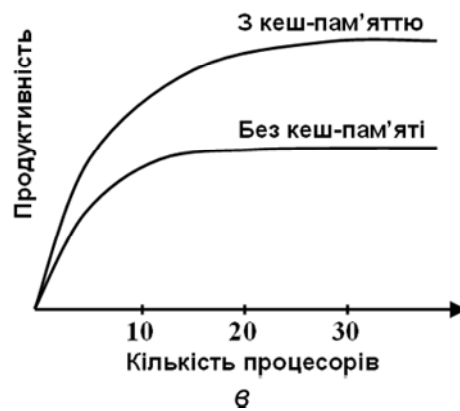
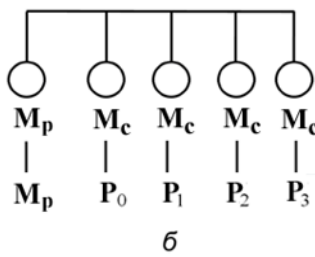
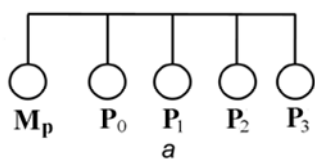
:

(. 4.7,)

11-

UMA

. 4.7, .



. 4.7.

UMA

4-8

32-64

NUMA (Non-Uniform Memory Access).

(. 4.3,)

(> 0,9)

(. 4.3,),

20

30

NUMA

, CC-NUMA NCC-NUMA.

(, Cache Only Memory

Architecture)

« »

NUMA.
Cache Coherent Non-Uniform Memory Architecture)

CC-NUMA

(NCC-NUMA, Non-Cache Coherent Non-Uniform Memory Architecture)

CC-NUMA

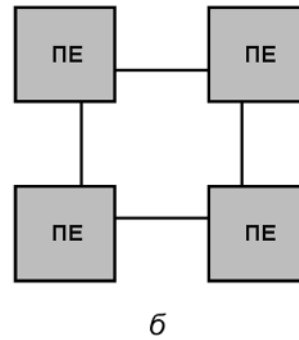
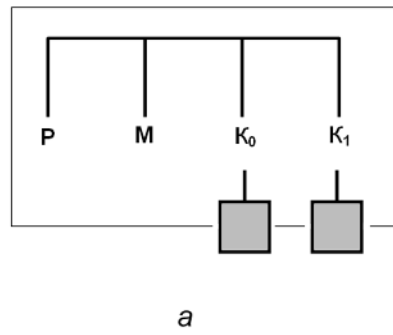
memory architectures).
 (virtual shared
 CC-NUMA,
 DSM.
 NUMA,

MIMD,

4.2.3

(NORMA, No Remote Memory Access).

, , :
 , , . - ,
 - ' .
 ,
 - ' , .
 ,
 , .
 , , :
 • , ;
 •
 .
 , . 4.4.
 (4.8,) ().
 (), ()
 / (0 K_1). (4.8,)
 , , .
 / .
 ,
 ,
 / .
 , - .



.4.8.

— ; — ’

’ ;

,

’ (DSM, Distribute Shared

Momory),

,

(SC-NUMA,

Software-Coherent Non-Uniform Memory Architecture).

’ ,

’ ,

’ . ,

’ ,

’

« » ’

.

4.2.4

— ’

’ ,

’

’ .

’ .

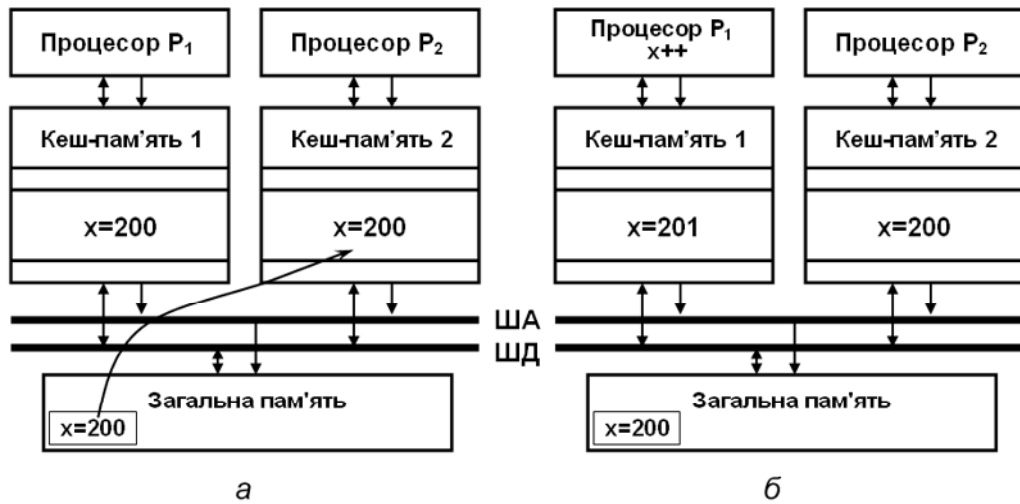
’ ,

’ ,

.

— ’ ,

, . ,
 - ,
 - ' .
 , - ,
 - , ,
 , ,
 .
 - ' ,
 (. 4.9). 1 2 ' ,
 , .
 , , ,
 (.4.9,). 1
 .
 - ' , -
 - ' 1. 2
 , - 2
 « » , - ' (.
 4.9,).
 , , ,
 , , ,
 - ' .
 .
 . ,
 .



.4.9.

(write through)

(write back).

(write invalidate)

(write update).

(write broadcast).

0.

4

1. , - .
- 2.
3. .
4. .
5. , , .
,
6. ,
7. , : UMA, NUMA.
8. , : COMA, CC-NUMA, NCC-
NUMA. .
9. , .
10. - ,
11. , - ,
,

5.

5.1.

5.1.1

,
,
,
.
,
,
()
-
:

$$= T/(T+t), \quad (5.1)$$

t —
.
,
 $= 1 -$.
-
,
-
.
,
,
,
« »
.
,
.
,
,
,

— .
,
” , , , , ,
.
,
—
.
—
,
.
.
, , ,
,
:
• —
;
• , —
;
• — RAID ,
(,) ,
, ;
• — ,
, ,
.
,
— ,
,

(')
0.99.

4

0.9999,

5.1.2

C

b

($b \in$),

; b

(\in

),

— , .

,

, .

, (,)

— ,

, .

— ,

, .

, ,

, .

— ,

, ,

— , ,

, ,

— , ,

, ,

— , ,

, ,

, ,

, ,

— , ,

, .

— ,

, .

-
 , .
 : () ;
) ;
 () ;
 (-) -
 ,) .
 () - ,
 () ,

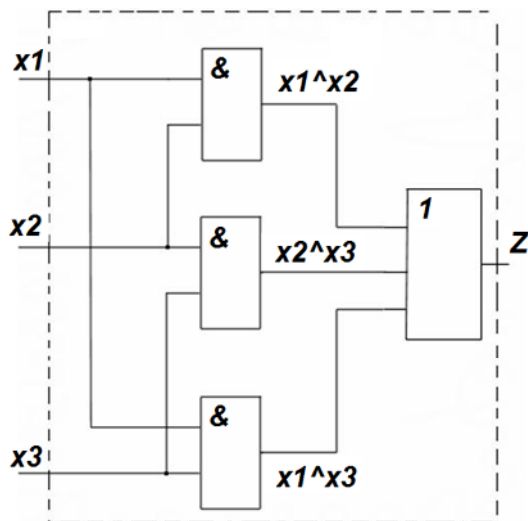
:

$$(\) = 1 - [1 - (\)]^{+1}, \quad (5.2)$$

- , .
 , , 0,7
 0,91, - 0,973.

:
 , $> k$, $k -$
 .
 ,
 .
 -
 .
 - «
 »
 .
 .
 :
 ;
 , , .

3».



.5.1

«2 3»

5», «4 7» . . .
 3»

«3

«2

. 5.1

$$: Z = x_1 x_2 + x_2 x_3 + x_1 x_3.$$

-
, , .
,
, .
, .
, .
,
, .
.
.
.
- .
- .
,
.
.
.

5.2.

5.2.1

() - ,
.
, , , ,
, , , ,
, , , ,
, , , ,
() :
- ,
;
- .
:
- ;
- ;
- .

5.2.2

() , .
, :
) ,
();
)
,
- 0,3;

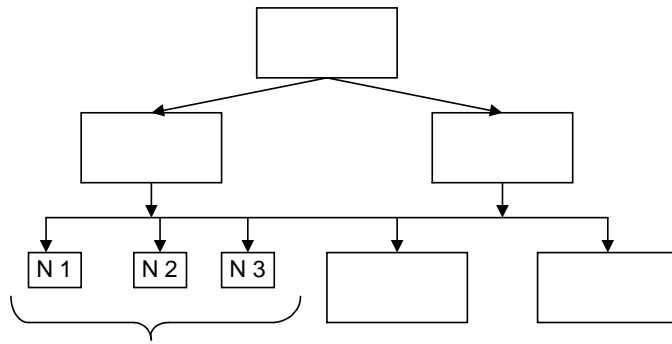
)

;

)

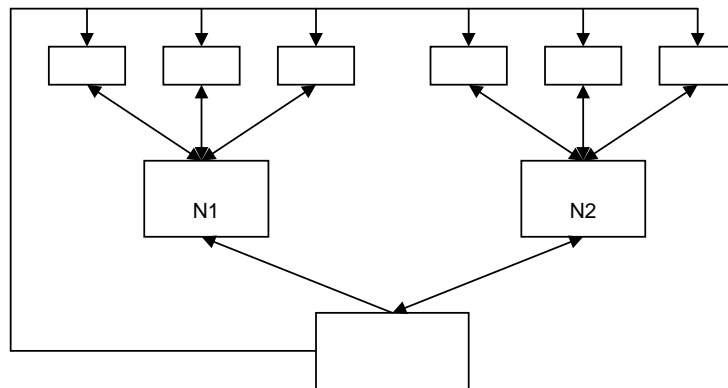
5.2.3

. 5.2.



. 5.2

. 5.3.



. 5.3.

,
().

.

-

.

.

,

()

,

,

,

,

,

.

()

.

:

) ;

) ;

) ;

) ;

) ;

) ;

) ;

) ;

.

5.2.4

,

.

.

,

.

.

,

.

.

:

)

;

)

;

)

-

().

5.3.

5.3.1

:

- 1) - ;
- 2) ;
- 3)

,

,

.

,

-

,

,

.

5.3.2

, , :

- ;
-

;

•

(

,

,

)

,

;

•

-

,

(heartbeat),

: 1)

2)

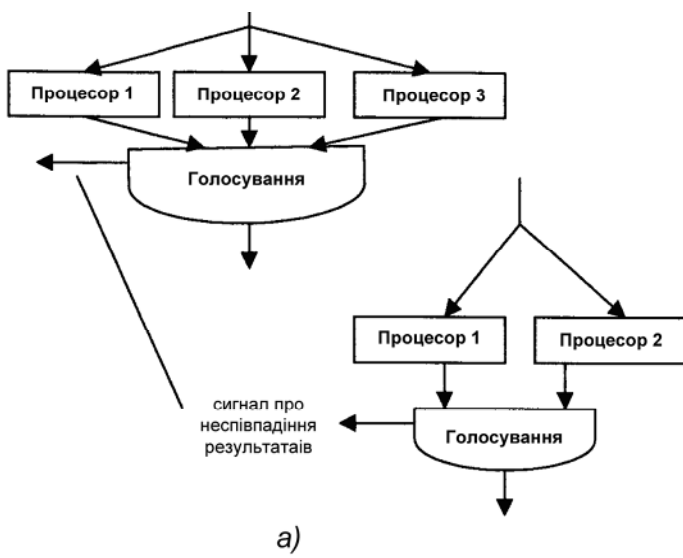
. 5.4,

, Tandem

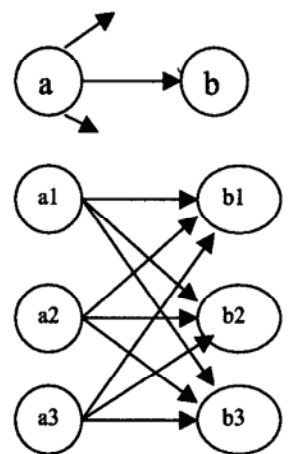
. 5.4,

b,

b_i , $b, i=1, 2, 3.$



a)



б)

. 5.4

n

n^2

. 5.5.

$B_i, i=1, 2, 3,$

a_i

. 5.5. .

1

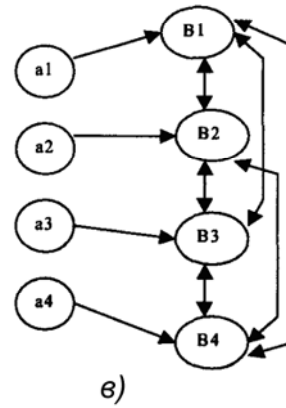
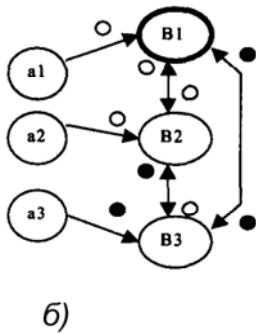
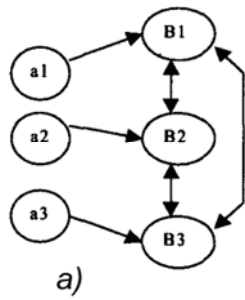
5.5. .

2

. 5.5. .

«

».



.5.5

5.3.3

:

1)

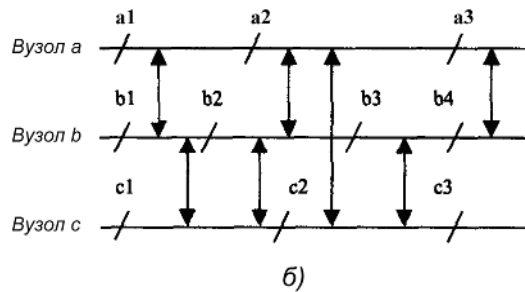
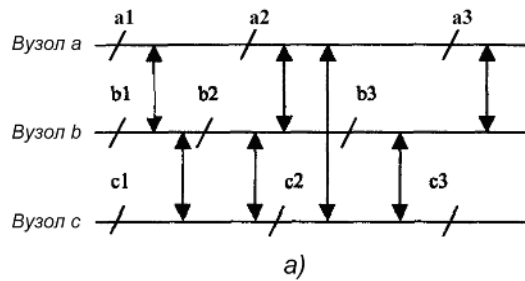
;

2)

3.3.
 5.6, — $a_1, 2, 3,$ — $b_1, b_2, b_3,$
 — $c_1, 2, 3.$

3
 — b
 3,
 — $b.$
 — b $b_3,$

« ».



.5.6

,
,
.

, 3, b4, 3 .5.6, .

5

1. .
- 2.
3. ?
4. ?
5. ?
6. ?
7. ?
8. ?
9. .
10. ?
11. « »?

1. –
, 2004. – 608 .
2.
. - - ;
. , 2007.
3. – , 1980 – 520 .
4. , C++.
– : – 2002.
4. – « » ,
1999., - 320 .
5. ,
“ ”, , 2003
6. , –
, 2008. – 470 .
7.
. , 2002
8. :
. , 2002.
9. 5- / , –
; : BXV, 2003. – 848 .
10. :
. - : , 1990.
11. / , : , 1988.
12.
. - : , 1991.
13. , - « » , 2002.
14. / ,

