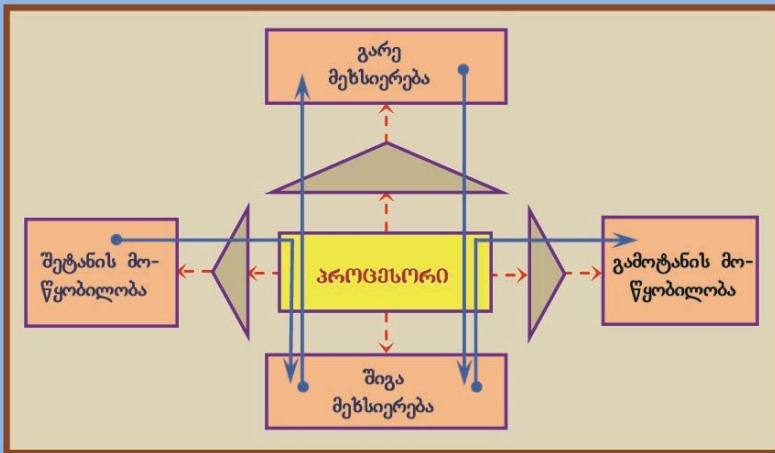


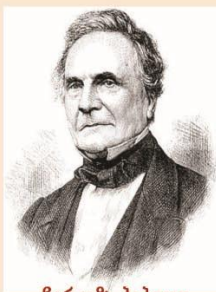
# კლასიკური დუნდუა

## კომპიუტერული სისტემების ტექნიკური საშუალებები (**HARDWARE**)



საგამომხეობლო სახლი  
„ტექნიკური უნივერსიტეტი“

## კომპიუტერული სისტემის უმჯობეს პიონერები



ჩარლზ ბებიჯი  
(1791 – 1871)



ადა ლავლეისი  
(1815 – 1852)



ჯონ ათანასოვი  
(1903 – 1995)



ალან ტიურინგი  
(1912 – 1954)



ჯონ ფონ ნეიმანი  
(1903 – 1957)



ჯონ მორლი  
(1907 – 1980)

საქართველოს ტექნიკური უნივერსიტეტი

ალექსანდრე ღუნდუა

კომპიუტერული სისტემების  
ტექნიკური საშუალებები  
(HARDWARE)



რეკომენდებულია საქართველოს  
ტექნიკური უნივერსიტეტის  
სარედაქციო-საგამომცემლო საბჭოს  
მიერ. 05.07.2019, ოქმი №2

თბილისი  
2019

შპს 004

დამხმარე სახელმძღვანელო შეიცავს კომპიუტერული სისტემების ტექნიკური საშუალებების ფორმირებისა და განვითარების საკვანძო საკითხებს, რომელთა შესწავლა გათვალისწინებულია სასწავლო დისციპლინით „კომპიუტერული სისტემები და გამოყენებითი ტექნოლოგიები“. მასში გადმოცემულია კომპიუტერული სისტემების წარმოქმნის წინაპირობები, ფორმულირებულია კომპიუტერის აგების კლასიკური პრინციპები და მათი ევოლუციური განვითარების გზა. დიდი ადგილი აქვს დათმობილი თანამედროვე პერსონალური კომპიუტერების ტექნიკური აღჭურვილობის, სტრუქტურული აგებულებისა და ფუნქციონირების თავისებურებების განხილვას.

წიგნი განკუთვნილია სატრანსპორტო და მანქანათმშენებლობის ფაკულტეტის „ტრანსპორტის“, „სავაზო ინჟინერიისა“ და „ბიზნესის ორგანიზაციისა და მართვის“ საგანმანათლებლო პროგრამათა ბაკალავრებისათვის. შეიძლება გამოყენებული იქნეს ტექნიკური პროფილის მაგისტრანტების, სპეციალისტებისა და კომპიუტერული სისტემებით დაინტერესებულ ადამიანთა ფართო წრისათვის.

რეცენზენტები: საქართველოს ტექნიკური უნივერსიტეტის  
ენერგეტიკისა და ტელეკომუნიკაციის ფაკულტეტის  
პროფესორი სერგო დადუნაშვილი,

საქართველოს ტექნიკური უნივერსიტეტის  
სატრანსპორტო და მანქანათმშენებლობის ფაკულტეტის  
ასოცირებული პროფესორი მურთაზ პაპასკირი

© საგამომცემლო სახლი „ტექნიკური უნივერსიტეტი“, 2019

ISBN 978-9941-28-513-4

<http://www.gtu.ge>

ყველა უფლება დაცულია. ამ წიგნის არც ერთი ნაწილის (იქნება ეს ტექსტი, ფოტო, ილუსტრაცია თუ სხვა) გამოყენება არანაირი ფორმით და საშუალებით (იქნება ეს ელექტრონული თუ მექანიკური) არ შეიძლება გამომცემლის წერილობითი ნებართვის გარეშე. საავტორო უფლებების დარღვევა ისჯება კანონით.

წიგნში მოყვანილი ფაქტების სიზუსტეზე პასუხისმგებელია ავტორი/ავტორები. ავტორის/ავტორთა პოზიციას შეიძლება არ ემთხვეოდეს საგამომცემლო სახლის პოზიცია.



## ს ა რ ჩ ე ვ ი

<b>I ტავი. კომპიუტერული სისტემები</b> .....	7 - 74
1.1. საანგარიშო მანქანების შექმნის სათავეებთან.....	7
1.2. საანგარიშო მანქანის თვისობრიობის შეცვლა.....	16
1.3. პირველი გამოთვლელი მანქანები .....	23
1.4. გამოთვლელი მანქანების აგების კლასიკური პრინციპები .....	29
1.5. გამოთვლელი მანქანების კლასიფიკაცია. ტერმინ «კომპიუტერის» დამკვიდრება .....	41
1.6. თანამედროვე კომპიუტერების კლასიფიკაცია .....	46
1.7. კომპიუტერთა თაობები და მათი დამახასიათებელი თვისებები.....	55
<b>II ტავი. პერსონალური კომპიუტერი</b> .....	75 - 147
2.1. პერსონალური კომპიუტერის ისტორია და არქიტექ- ქტექტურა.....	75
2.2. პერსონალური კომპიუტერების კლასიფიკაცია.....	83
2.3. პერსონალური კომპიუტერების აგების მაგისტრალ- ურ-მოდულური პრინციპი .....	86
2.4. სისტემური დაფა (დედა-დაფა) .....	92
2.5. მიკროპროცესორი: ძირითადი ელემენტები და მა- ხასიათებლები.....	97
2.6. პერსონალური კომპიუტერის მეხსიერება.....	107
2.7. პერსონალურ კომპიუტერში ინფორმაციის შემტა- ნი მოწყობილობები.....	128
2.8. პერსონალურ კომპიუტერიდან ინფორმაციის გამო- მტანი მოწყობილობები.....	143
<b>ლიტერატურა</b> .....	148

# I ტაჭი

## კომპიუტერული სისტემები

### 1.1. საანგარიშო მანქანების შიქმნის სათაჯმებთან

*კომპიუტერი* წარმოდგენს მონაცემების სახით წარმოდგენილი ინფორმაციის ავტომატურად დამამუშავებელ სისტემას. *ინფორმაცია*, როგორც სამეცნიერო კატეგორია, ერთმანეთისაგან განსხვავებული უამრავი დისციპლინის (ფიზიკის, ბიოლოგიის, ფილოსოფიის, მათემატიკის, პოლიტოლოგიის, ისტორიის და ა. შ.) შესწავლის საგანია. აღნიშნულის მიუხედავად, მისი ზუსტი *განსაზღვრება* არ არსებობს და განსაზღვრების ნაცვლად გამოიყენება *ცნება* ინფორმაციის შესახებ. განსაზღვრებისაგან ცნება იმით განსხვავდება, რომ მეცნიერებისა და ტექნიკის სხვადასხვა სფეროში ცნება ისეა ფორმულირებული, რომ მისი შინაარსი მაქსიმალურად შეესაბამება კონკრეტული დისციპლინის შესწავლის საგანსა და გადასაწყვეტ ამოცანებს. ფილოსოფია იძლევა *ინფორმაციის ცნების* უზოგადეს ფორმულირებას, რომლის თანახმადაც იგი მატერიალური სამყაროსათვის დამახასიათებელი ასახვის პროცესის პროდუქტია.

*ასახვა* ეწოდება ინფორმაციისა მიღების, შენახვისა და გადაცემის უნარს. არსებობს ასახვის შემდეგი ოთხი სახე: შემეცნება, ფსიქიკა, გაღიზიანებადობა და ურთიერთზემოქმედების დამახსოვრება. *ადამიანისათვის* ასახვის ოთხივე სახეა დამახასიათებელი. *ცხოველები* ინფორმაციას ასახავს ფსიქიკის, გაღიზიანებადობისა და ურთიერთზემოქმედების დამახსოვრებით, *მცენარეები* - გაღიზიანებადობისა და ურთიერთზემოქმედების დამახსოვრებით, ხოლო არაცოცხალი საგნები მხოლოდ ურთიერთზემოქმედების დამახსოვრების გზით.

თავდაპირველად ადამიანი, როგორც გონიერი არსება (*Homo Sapiens*), დიდ დროს უთმობდა *რაოდენობითი ინფორმაციის* ასახვას, რისთვისაც მოიგონა რიცხვები და ჩამოაყალიბა თვლის გარკვეული

სისტემები. თვლის პროცესში მისი შრომის შესამსუბუქებლად მან დაიწყო სხვადასხვა სახის სამარჯვებების შექმნა. ასეთი სამარჯვებები აღმოჩენეს არქეოლოგებმა და მათ საანგარიშო დაფები, ანუ **აბაკები** ეწოდა (ძვ. ბერძ. *αβαξ*, ლათ. *abacus* - «**დაფა**»). პირველი ასეთი აბაკები ჩვენი წელთაღრიცხვამდე **VI – V** საუკუნეებით თარიღდება და ისინი **ძველ რომსა** და **საბერძნეთში** იყო დამზადებული. ისინი წარმოადგენდა ბრინჯაოს, ქვის ან სპილოს ძვლისაგან დამზადებულ დაფებს, რომელთა ღრმულებში თავსდებოდა კენჭები და მათი გადა-  
ნაცვლებით ხდებოდა თვლა. შემდეგ სხვადასხვა ფორმის საანგარი-  
შო დაფები იაპონიასა და ჩინეთშიც აღმოჩნდა და მათ შესაბამისად „**სერობიანი**“ და „**სუანპანი**“ ეწოდა.

შუა საუკუნეებში აბაკის გამოყენება აღადგინა და მისი გაუმ-  
ჯობესებული მოდელი დაამუშავა ლეგენდებით მოცულმა მეცნიერმა  
და საეკლესიო მოღვაწემ **ჰერბერტ ავრილიელმა** (დაახლოებით **946 - 1003**), რომლის სახელი მოხსენიებულია ცნობილი რუსი მწერლის **მ. ბულგაკოვის** რომანში „**ოსტატი და მარგარიტა**“.

საანგარიშო სამარჯვებების შექმნაში თავისი წვლილი შეიტანა შო-  
ტლანდიელმა მათემატიკოსმა **ჯონ ნეპერმა (1550- 1617)**, რომელმაც **1617**წელს გამოცემულ წიგნში გვაჩვენა ხის ჩხირებით გამრავლების პრინციპი. ასეთ ჩხირებს „**ნეპერის ჩხირები**“ ეწოდა.

**XVII** საუკუნის დასაწყისში, როდესაც მათემატიკამ მეცნიერება-  
ში საკვანძო ადგილი დაიკავა, სულ უფრო და უფრო იგრძნობოდა **საანგარიშო მანქანის(საანგარიშოების)** შექმნის აუცილებლობა. ას-  
ეთ მანქანას კალკულატორსაც უწოდებენ (ლათ. „**calculo**“ ითარგ-  
მნება, როგორც „**ვითვლი**“).

**საანგარიშოები** თანამედროვე ციფრული მოწყობილობების წინა-  
პარია. ისინი ერთეულების თვლის მეშვეობით ობიექტების რაოდენ-  
ობითი შეფასების საშუალებას გვაძლევს. გარესამყაროს ნებისმიერი  
ობიექტების რაოდენობითი შეფასებისათვის მარტო თვლის ოპერა-  
ცია არ არის საკმარისი. არსებობს ისეთი ობიექტებიც, რომელთა  
რაოდენობრივი შეფასებისათვის **გაზომვის** ოპერაციის ჩატარებაა  
აუცილებელი. შეუძლებელია რომელიმე მათგანის გადაჭარბებულად  
შეფასება. ცალკობითი გამოთვლის გარეშე თუ წარმოუდგენელია ელ-  
ემენტარული რაოდენობითი შეფასება, ხოლო ცნობილი მეცნიერის

დ. მენდელეევის (1834-1907) თანახმად «*მეცნიერება იწყება იქ, სადაც იწყება გაზომვა*». რაოდენობითი შეფასების ამ ორი (თვლისა და გაზომვის) სახეობის შესაბამისად ისტორიულად ჩამოყალიბდა გამოთვლების ორი სახის, კერძოდ, ანალოგური და ციფრული მიმართულება.

**ანალოგური მიმართულება** იყენებს პრინციპს, რომლის დროსაც რიცხვები მათი „ანალოგებით“ იცვლება. მაგალითად, **ლოგარითმული მეთოდის** დროს რიცხვები იცვლება მონაკვეთების სიგრძეებით. ასეთი ანალოგი არაა **დისკრეტული**: იგი არ იზრდება რიცხვის უმცირესი თანრიგის სიდიდის ტოლი ბიჯებით. იგი უწყვეტი სიდიდეა, რომლის გაზომვისას აუცილებლად წარმოიშვება გარკვეული ცდომილობა, რაც მის ზუსტად შეფასებას შეუძლებელს ხდის; გარდა ამისა, ამ მეთოდის პრაქტიკული გამოყენება ხშირად ტექნიკურადაც მოუხერხებელია; მაგალითად, ლოგარითმული სახაზავით რომ დავამუშავოთ, ვთქვათ, **10**-თანრიგიანი რიცხვები, საჭიროა სახაზავის სიგრძე რამდენიმე ათეული მეტრის ტოლი იყოს. ცხადია, რომ ასეთი პროექტის რეალიზაცია აბსოლუტურად უაზრობაა.

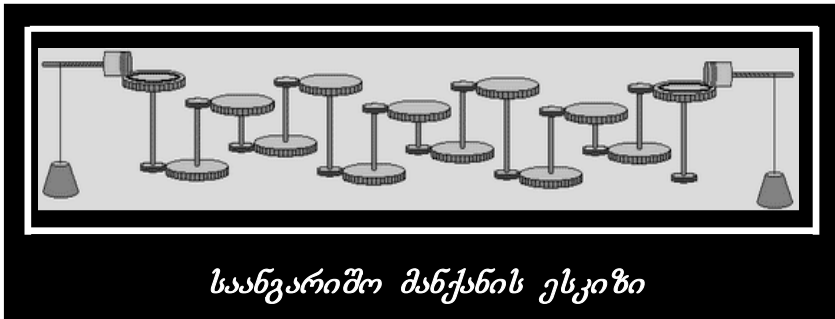
გამოთვლების ანალოგური მიმართულების ფუძემდებელი იყო შოტლანდიელი მათემატიკოსი **ჯონ ნეპერი (1550-1617)**. გამოთვლის ეს რეკოლუციური მეთოდი მან გადმოსცა **1614** წელს ლათინურ ენაზე გამოქვეყნებულ ნაშრომში «*Mirificilogarithmmorum canonicis descriptio*» («ლოგარითმების საოცარი ცხრილების აღწერა/*Description of the amazing table of logarithms*»). ლოგარითმულ კანონზე დაყრდნობით, რომელიც გამრავლებისა და გაყოფის ოპერაციებს „ცვლიდა“ შეკრებისა და გამოკლების ოპერაციებით, შედგენილი იქნა რიცხვების ვეებერთელა მასების გამომყენებელი ასტრონომების მუშაობის შემამსუბუქებელი ცხრილები. მოგვიანებით ნეპერის რეკომენდაციით ინგლისელმა მათემატიკოსმა **ჰენრი ბრივსმა (1561-1630)**, ლოგარითმის გამოსახულება მის თანამედროვე სახეს დაუახლოვა.

ლოგარითმების თეორიისა და ცხრილების საფუძველზე შეიქმნა პირველი ლოგარითმული სახაზავები. **ლოგარითმული სახაზავი** წარმოადგენს **ანალოგურ გამოთვლელ მოწყობილობას**, რომელიც რამდენიმე მათემატიკური ოპერაციის (გამრავლების, გაყოფის, ახარისხების, ტრიგონომეტრიული ფუნქციების გამოთვლის, გალოგარით-



მების და ა. შ.) შესრულების საშუალებას იძლევა. შემდგომ დამუშავდა ლოგარითული სკალიანი სხვადასხვა მოწყობილობა და სახაზავი, ხოლო დაახლოებით ასი წლის შემდეგ გამოჩნდა დღეს კარგად ცნობილი **ლოგარითული სახაზავი**. მეოცე საუკუნის 30-იან წლებამდე, მანამ, სანამ არ დამუშავდა ელექტრული არითმომეტრები, გრძელდებოდა ამ სახაზავის ბატონობა.

ინტეგრალების, დიფერენციალებისა და ა. შ. გამოთვლის ოპერაციათა სირთულემ განაპირობა ანალოგური მოწყობილობების მთელი კლასის გაჩენა; ისინი განკუთვნილი იყო მათემატიკის საკითხებში არც თუ ისე გაწაფული მომხმარებლების მიერ კონკრეტული მათემატიკური გამოთვლების შესასრულებლად. კერძოდ, XIX საუკუნეში დამუშავდა შემდეგი ანალოგური მოწყობილობები: **პლანიმეტრი** – ბრტყელი ფიგურების ფართობის გამოსათვლელად, **კურვიმეტრი** – მრუდეების სიგრძის განსასაზღვრავად, **დიფერენციატორი** – წარმოებულების გამოსათვლელად, **ინტეგრატორი** – ინტეგრალების განსასაზღვრავად, **ინტეგრაფი** – გრაფიკულად მოცემული ფუნქციათა ინტეგრალების გრაფიკული ფორმით გამოსათვლელად, **ინტეგრირმეტრი** – გრაფიკულად მოცემული ფუნქციების დასამუშავებლად.



**ნახ. 1.1.** ლეონარდო და ვინჩის საანგარიშო მანქანის ესკიზი

**ციფრული მიმართულება** გაცილებით პერსპექტული აღმოჩნდა და იგი საფუძვლად დაედო გამოთვლით ტექნიკას. არსებული ისტორიული მასალების მიხედვით მის განვითარების ეტაპს საფუძველი ჩაეყარა 1492 წელს, როდესაც იტალიელმა ფერმწერმა, არქიტექტორმა და ინჟინერმა **ლეონარდო და ვინჩიმ (1452-1519)** დაამუშავა საანგარიშო მანქანის ესკიზი (ნახ. 1.1) და იგი დაწვრილებით აღწე-

რა თავის დღიურებში, რომელიც დღეს „*მადრიდული კოდექსის*“ სახელწოდების ორტომეულის სახითაა ცნობილი. ეს დღიურები **1967** წლამდე, მანამ სანამ იგი არ იპოვეს ამერიკელმა მკვლევრებმა, უცნობად იყო „ესპანეთის ბიბლიოთეკაში“. აღნიშნული ესკიზის თანახმად საანგარიშო მანქანას უნდა ჰქონოდა პატარა თვლებიანი ღეროები. მექანიზმს უნდა ემუშავა თვლებზე არსებული კბილების წყალობით. კერძოდ, პირველი თვლის ღერძს **10** ბრუნით შემობრუნების შემდეგ ერთი ბრუნით შემობრუნებინა მეორე თვალს, მეორე თვალი **10** ბრუნის შემდეგ – მესამე თვალი და ა. შ. დღიურის თანახმად, *ლეონარდო და ვინჩი* ვერ შეძლო თავისი ჩანაფიქრის პრაქტიკული რეალიზება, მისი ესკიზის საფუძველზე **IBM** ფირმამ ააგო კბილანებისაგან შემდგარი სრულიად მუშაობის უნარიანი **13**-თანრივიანი ამჯამავე მანქანა.

ციფრულმა საანგარიშო მანქანებმა დაუღო სათავე თანამედროვე კომპიუტერულ სისტემებს, ამიტომ მათ უფრო დაწვრილებით განვიხილავთ.

ბევრი ავტორი დღემდე არასწორად თვლის, რომ პირველი მექანიკური საანგარიში მანქანა **1642** წელს შექმნა ცნობილმა ფრანგმა მეცნიერმა *ბლეზ პასკალმა*. კეპლერის სამეცნიერო ცენტრის დირექტორმა *ფრანც ჰემერმა* შტუდგარდტის (გერმანია) ბიბლიოთეკაში მუშაობისას **1957** წელს შემთხვევით აღმოაჩინა საანგარიშო მანქანის მანამდე უცნობი ესკიზის ფოტოასლი (დედანი პეტერბურგთან ახლომდებარე *ჰულკოვოს ობსერვატორიაში* არსებულ *იოჰან კეპლერის არქივში ინახება*). მან დაადგინა, რომ ეს ესკიზი წარმოადგენდა *ტიუბინგის უნივერსიტეტის* პროფესორის *ვილჰელმ შიკარდის (1592–1635)* მიერ *კეპლერისადმი 1624* წლის **25** თებერვალს გაგზავნილი წერილის დანართს, რომელშიც *შიკარდი* აღწერდა მის მიერ გამოგონებულ საანგარიშო მანქანას. მანქანა შეიცავდა ამჯამამდე და მამრავლებელ მოწყობილობებს, აგრეთვე საშუალებად შედეგების ჩამწერ მექანიზმს. მამასადამე, *შიკარდის მანქანად* წოდებული პირველი საანგარიშო მანქანა *ბლეზ პასკალზე 19* წლით ადრე – **1623** წელს – გამოიგონა *შიკარდმა*.

*შიკარდის* მანამდე უცნობი მანქანის ესკიზის აღმოჩენა სრულიად არ ამცირებს ცნობილი ფრანგი ფიზიკოსის *ბლეზ პასკალის (1623-1662)* დამსახურებას, რომელმაც ჯერ კიდევ **19** წლის ასა-

კში გადასახადების ამკრებად მომუშავე მამის შრომის შესამსუბუქებლად **1642** წელს სრულიად დამოუკიდებლად დაამუშავა და ააგო მექანიკური ამჯამავი მოწყობილობა. „**პასკალინად**“ წოდებული აღნიშნული მოწყობილობა წარმოადგენდა ერთმანეთთან დაკავშირებული მრავალი კბილანას მქონე მექანიკურ მანქანას. შესაკრები რიცხვები მანქანაში ამკრეფი თვლების სათანადოდ შებრუნების გზით შეიტანებოდა. თვლების რაოდენობა შესატანი ათობითი რიცხვის თანრიგების რაოდენობას უდრიდა; თითოეულ თვალს შეესაბამებოდა გარკვეული თანრიგი და მასთან დატანილი იყო **0**-დან **9**-მდე ციფრებით დანომრილი დანაყოფი. რიცხვის შესატანად საჭირო იყო კბილანა სათანადო ციფრამდე შეგვეტრიალებინა.

**1673** წელს გერმანელმა ფილოსოფოსმა და მათემატიკოსმა **გოტფრიდ ლაიბნიცმა (1646-1716)** შექმნა ძირითადი არითმეტიკული მოქმედებების შემსრულებელი მექანიკური კალკულატორი „**საფეხურხუროვანი გამომთვლელი**“. იგი **20** წლის განმავლობაში აუმჯობესებდა საკუთარ საანგარიშო მანქანას. ამის შედეგად მიღებულ მოდელს შეეძლო **8**-თანრივიანი რიცხვების შეკრება, გამრავლება, გაყოფა და ახარისხება. გამრავლებისა და გაყოფის შედეგს ჰქონდა **16** თანრიგი. ლილვების მახრუნი სახელურისა და გამრავლების პროცესში წარმოქმნილი კერძო გამრავლების დაჯამებით მიღებული რიცხვის ავტომატურად მაკონტროლებელი მექანიზმის წყალობით მანქანა ძალიან მარტივად იმართებოდა. აღნიშნული მანქანის უმთავრესი ღირსება მაინც ისაა, რომ ლაიბნიცმა არითმეტიკული მოქმედებებისათვის თვლის ჩვეულებრივი **ათობითი სისტემის** ნაცვლად პირველად გამოიყენა თვლის **ორობითი სისტემა**.

**ლაიბნიცმა** პირველად შეიქნო თვლის ორობითი სისტემის მნიშვნელობა და როლი. **1679** წლის მარტში ლათინურ ენაზე შედგენილ ხელნაწერში ლაიბნიცმა ახსნა გამრავლების შესასრულებლად თუ როგორ შეგვეძლო გამოგვეყენებინა თვლის ორობითი სისტემა. მოგვიანებით მან დაამუშავა თვლის ორობითი სისტემის გამომყენებელი გამომთვლელი მანქანის პროექტი. მასში იგი წერდა, რომ ორობითი სისტემა საშუალებას გვაძლევს „... თვლის პროცესი ყოველგვარი დამატებითი დანახარჯების გარეშე შევასრულოთ. ამისათვის საკმარისია პატარა ბურთულებიან კოლოფის ძირზე დავი-

ტანოთ ნახვრეტები, რომელთა გაღება და დახურვა შეგვეძლება. ორობითი რიცხვის წარმოსადგენად გავაღოთ  $1$ -ის და დახუროთ  $0$ -ის შესაბამისი ნახვრეტები; კოლოფის ქვემოთ კი მოვათავსოთ სვეტების შესაბამისი სპეციალური ღარები. ღია ნახვრეტებიდან ღარში ჩავარდება კოლოფში არსებული პატარა ბურთულები, ხოლო დახურული ნახვრეტებიდან არაფერი არ ჩავარდება. კოლოფი უნდა ვამოძრაოთ და სვეტიდან სვეტზე (ღარიდან ღარზე) ისე გადავანაცვლოთ, როგორც ამას მოითხოვს გამრავლების წესი; ამასთანავე, მანქანის მუშაობის დაწყებამდე ვერც ერთი ბურთულა ვერ უნდა მოხვდეს ერთი ღერიდან მეორეში ... „. ორობითი არითმეტიკის საკითხები ლაიბნიცმა დაწვრილებით განიხილა მოგვიანებით (1703 წელს) გამოქვეყნებულ ტრაქტატში „*Explication de l'Arithmetique Binaire*“. *მეოცე საუკუნის შუა რიცხვებში გამოგონებულ კომპიუტერებში თვლის ორობითი სისტემის გამოყენებისას, სამწუხაროდ, სათანადოდ არავის არ შეუფასებია ამ მიმართულებით ლაიბნიცის მიერ გაწეული ღვაწლი.* (<http://infhistory.com/lejbnic.html>).

XVII საუკუნეში, რა თქმა უნდა, შეუძლებელი იყო ლაიბნიცის კალკულატორების სერიული გამოშვება, მაგრამ ისინი არც ისე მცირე რაოდენობით იყო დამზადებული.

1709 წელს იტალიელმა მათემატიკოსმა, ფიზიკოსმა და ასტრონომმა *ჯოვანი პოლენიმ (1683—1761)* მოახდინა პირველი არითმომეტრის (ბერძ. *Arithmos* - რიცხვი) დემონსტრირება. ეს იყო მართივი არითმეტიკული მოქმედებების (შეკრების, გამოკლების, გამრავლებისა და გაყოფის) შემსრულებელი სახელურიანი სამაგიდო მექანიკური საანგარიშო მანქანა. მასში გამოყენებულ სიახლეებს შორის აღსანიშნავია ის, რომ მანქანას ამოქმედება ბავირის თავისუფალ ბოლოზე მიბმული მვარდნი ტვირთის ძალა. არითმომეტრების აგების ისტორიაში ეს იყო ენერჯის წყაროთი ხელის ამძრავის შეცვლის პირველი მცდელობა.

1723 წელს გერმანელმა ფიზიკოსმა და ასტრონომმა *ქრისტიან ლუდვიგ ჰერსტენმა (1701- 1762)* ლაიბნიცის ნაშრომების საფუძველზე დაამუშავა და 1725 წელს ააგო *არითმეტიკული მანქანა*, რომელიც საშუალებას გვაძლევდა გამოგვეთვალა გაყოფის შედეგად წარმოქმნილი წილადური ნაწილი (ნაშთი) და მიმდევრობით შეგვეკრიბა გამრავლების პროცესში ფორმირებული შესაკრებები. *ჰერს-*

*ტენის მანქანაში* მეორე შესაკრების შეტანის სისწორის კონტროლიც იყო გათვალისწინებული, რაც გამომთვლელის გადაღლით გამოწვეული სუბიექტური შეცდომის დაშვების აღბათობას ამცირებდა. მათემატიკასა და ფიზიკაში მიღწეული წარმატებების გამო **1733** წელს იგი *დიდი ბრიტანეთის სამეფო საზოგადოების* წევრად აირჩიეს

**1783** წელს გერმანელმა ინჟინერმა და *არქიტექტორმა იოჰან მიულერმა (1746-1830)* დაამუშავა **14** თანრიგიანი საანგარიშო მანქანის კონსტრუქცია და იგი ქალაქ *ჰესენში* დაამზადებინა საათების ოსტატს; მასში ჩაშენებული იყო ზარი, რომელიც გამომთვლელის მიერ გარკვეული სახის შეცდომის დაშვებისას სიგნალს იძლეოდა. **1786** წელს მან წამოაყენა საანგარიშო მანქანის ასაგებად „*სასრული სხვაობის*“ მეთოდის გამოყენების იდეა, რომლის რეალიზება მოგვიანებით მოახდინა ცნობილმა ინგლისელმა მეცნიერმა *ჩარლზ ბეებიჯმა* და ააგო საეტაპო მნიშვნელობის „*სხვაობითი მანქანა*“. სწორედ ამ უკანასკნელმა შეცვალა საანგარიშო მანქანების თვისობრიობა და სათავე დაუდო *ავტომატური გამომთვლელი მანქანების* შექმნას; ამიტომ მომდევნო პარაგრაფში ჩვენ მეტ-ნაკლებად ვრცლად განვიხილავთ მას.

განსაკუთრებით აღნიშვნის ღირსია ფრანგი გამომგონებელის *ჟოზეფ ჟაკარდის (1752-1834)* გამოგონება, რომელმაც დიდი როლი ითამაშა გამომთვლელი მანქანების შექმნაში. კერძოდ, მან **1804** წელს შემოგვთავაზა მომუშავე საქსოვ დაზგაზე ძაფის ავტომატური კონტროლის ხერხი. ამისათვის *ჟაკარდმა* გამოიყენა ქსოვილზე დასატანი მოხატულობის შესაბამისად გარკვეულ ადგილებზე დანერგტილი სპეციალური ბარათები – *პერფობარათები*. ჩარხის მუშაობა დაპროგრამდებოდა პერფობარათების დასტის დახმარებით, რომელთაგანაც თითოეული ბარათი მართავდა ჩარხის ერთ მაქოს. სხვა მოხატულობაზე გადასასვლელად საკმარისი იყო ოპერატორს უბრალოდ შეეცვალა პერფობარათების დასტა. მოგვიანებით სწორედ პერფობარათები იქნა გამოყენებული გამომთვლელი მანქანისათვის საჭირო მონაცემებისა და ბრძანებების შესანახად.

**1820** წელს ფრანგმა მეწარმემ *ჩარლზ-ქსავერ თომას კოლმარმა (1785-1870)* შექმნა პირველი კომერციული არითმომეტრი (თავდაპირველად იგი საკმაოდ ძვირი – **400** ფრანკი ღირდა), რომელიც ავ-

ებული იყო *ლაიბნიცის კალკულატორის* პრინციპის მიხედვით. მას შეეძლო **30**-თანრიგიან რიცხვების დამუშავება და ამით ჯობდა ყველა არსებულ საანგარიშო მანქანას.

## 1.2. საანგარიშო მანქანის თვისობრიობის უმცვლა

*XIX* საუკუნემდე საანგარიშო მანქანები იქმნებოდა აბსტრაქტული არითმეტიკული ოპერაციების შესასრულებლად. საანგარიშო მანქანების აგების განსხვავებული სტრატეგია აირჩია ინგლისელმა მათემატიკოსმა და გამოგონებელმა **ჩარლზ ბებიჯმა (1791-1871)**. ეპოქაში, როდესაც იგი სამეცნიერო ასპარეზზე გამოვიდა, ევროპაში ფართოდ იყო გავრცელებული სხვადასხვა სახის, კერძოდ, არითმეტიკული, ტრიგონომეტრიული და ლოგარითმული ცხრილები. მათ საზოგადოებრივი ცხოვრების ერთმანეთისაგან დიამეტრულ სფეროებში გამოიყენებდნენ. ბანკებსა და სასესხო კანტორებში გამოიყენებოდა პროცენტების ცხრილები, სადაზღვევო კომპანიებში – სიკვდილიანობის აღმრიცხველი ცხრილები, საზღვაო საქმიანობაში – ასტრონომიული და სანავიგაციო ცხრილები. ეს უკანასკნელები მეტად მნიშვნელოვანი იყო ინგლისისათვის, რომელიც იყო უდიდესი საზღვაო სახელმწიფო. **1776** წლიდან იქ დაიწყო „**საზღვაო კალენდრის**“ ყოველწლიური გამოცემა, რომელშიც დიდი ადგილი ეკავა ლოგარითმული ცხრილებს. მათ შედგენაზე მუშაობდა მრავალი გამოთვლელი. ცხრილებში არსებული მონაცემების სისწორეზე დიდად იყო დამოკიდებული გემების უსაფრთხო მოძრაობა. ვინაიდან ცხრილები ასობით და ათასობით შეცდომებს შეიცავდა, ამიტომ მათ აღმოჩენისა და გასწორებისათვის გამოიმკვმლები იძულებული იყვნენ შეენახათ კორექტორების მთელი შტატი, მაგრამ საქმეს არც ეს შეელოდა.

ცნობილი ფრანგმა მათემატიკოსის **გასპარ კლერ ფრანსუა მარი რიშის** (ბარონ **დე პრონის, Gaspard Clair François Marie Riche, baron de Prony, 1755-1839**) მიერ გამოსავლის პოვნის მიზნით საზოგადოებას შესთავაზა გამოთვლების „კონვეიერული სისტემა“; მან გამოთვლები დაყო **3** ჯგუფად; **პირველ ჯგუფში** დასაქმდა **5-6**

ცნობილი მათემატიკოსი, რომელთა შორისაც იყო ცნობილი ფრანგი მათემატიკოსი **ადრიენ მარი ლეჟანდრი (Adrien-Marie Legendre 1752-1833)**; ისინი ირჩავდნენ სანგარიშოდ ყველაზე მეტად მოსახერხებელ მეთოდებს, ფორმულებს და ადგენდნენ გაანგარიშებათა სქემებს; **მეორე ჯგუფში** შეიყვანეს მაღალკვალიფიციური **7-8** კამომთვლელი, რომლებიც შერჩეული ფორმულების გამოყენებით **5** ან **6**-ბიჯიანი ინტერვალებით განსაზღვრავდნენ ფუნქციათა რიცხვით მნიშვნელობებს; მესამე ჯგუფში შეიყვანეს დაბალი კვალიფიკაციის **90**-მდე კამომთვლელი, რომლებიც ავსებდნენ ცხრილებში მეორე ჯგუფის კამომთვლელების მიერ დატოვებულ ცარიელ ინტერვალებს.

**ჩარლზ ბებიჯმა**, რომელიც ძალიან დიდ პატივს სცემდა **ლეჟანდრის**, წამოაყენა წინადადება კამომთვლელების მესამე ჯგუფი შეეცვალათ საანგარიშო მანქანით, რომელიც არა აბსტრაქტული არითმეტიკული ოპერაციების შესრულებით იქნებოდა დაკავებული, არამედ მოახდენდა, როგორც იგი ამბობდა, „**ადამიანური ინტელექტის უაღრესად პრიმიტიულ ქმედებათა ავტომატიზებას**“.

**ბებიჯის მიერ** შემოთავაზებული მანქანა განკუთვნილი იყო მრავალწევრების ტაბულაციისათვის. ამისათვის მან ამოიჩნია **რიცხვით ანალიზში** იმ პერიოდისათვის კარგად ცნობილი **სხვაობათა ხერხი** (ამ ხერხის გამოყენების იდეა, როგორც ზემოთ აღვნიშნეთ, ჯერ კიდევ **1786** წელს წამოაყენა გერმანელმა მათემატიკოსმა **ი. მიულერმა**). სხვაობათა ხერხი განვიხილოთ შემდეგი მარტივი მაგალითის საფუძველზე: გამოვითვალოთ ნატურალური რიცხვების მწკრივის **მეოთხე ხარისხთა ცხრილი**, ე.ი. მოვახდინოთ  $N = n^4$  ( $n = 1, 2, \dots$ ) ფუნქციის ტაბულირება (ნახ. **1.2**).

დავუშვათ, რომ  $n = 1, 2, \dots, 6$ -ისათვის გამოთვლილია  $N$ -ის მნიშვნელობები (იხ. **1.2** ნახაზის „ბ“ სვეტი) და გვინდა გამოვითვალოთ მისი დანარჩენი მნიშვნელობები.

„ბ“ სვეტში მეორე ადგილიდან დაწყებულ ყველა წევრს გამოვაკლოთ წინა წევრი და მიღებული შედეგები აღვნიშნოთ  $\Delta_1$  სხვაობებით. მათ შეესაბამება **1.2** ნახაზის შავფონიანი „ბ“ სვეტი. ამ უკანასკნელში არსებულ რიცხვებზე თუ შევასრულებთ იმავე ოპერაციებს, მივიღებთ  $\Delta_2$  სხვაობას, რომელიც **1.2** ნახაზის შავფონიანი

„ღ“ სვეტის სახითაა ჩამოწერილი. ანალოგიურად მიიღება  $A_3$  და  $A_4$  სხვაობებიანი „მ“ და „მ“ სვეტები. ამ უკანასკნელ სვეტში მოთავსებული  $A_{n=4}$  სხვაობები მუდმივია და ისინი  $24$ -ის ტოლია. ეს შემთხვევით არ მომხდარა – იგი უმნიშვნელოვანესი თეორემის შედეგია, რომლის თანახმადაც  $n$ -ური ხარისხიანი მრავალწევრის  $n$ -ური სხვაობები მუდმივია.

შავ ფონზე გამოსახული **1.2** ნახაზის შედგენის შემდეგ დანარჩენი  $n$ -ების ( $n = 7, 8, \dots$ ) მეოთხე ხარისხები შეიძლება გამოითვალოს ცხრილის მარჯვენა კიდედან დაწყებული შეკრების ოპერაციების საშუალებით. **1.2** ნახაზზე გამოთვლილია  $7^4$ -ისა და  $8^4$ -ის მნიშვნელობები, კერძოდ:  $7^4 = 2041$  და  $8^4 = 4096$

**ჩარლზ ბებიჯმა** შეკრების ოპერაციებით  $n$ -ური მრავალწევრების მნიშვნელობათა გამოთვლელი მანქანის აგების მიზანი დაისახა და მას *სხვაობითი მანქანა* უწოდა. აქედან ნათლად ჩანს ადრეულ *საანგარიშო მანქანებსა* და ბებიჯის *სხვაობით მანქანას* შორის არსებული განსხვავება. *სხვაობითი მანქანა* უკვე არ უნდა ყოფილიყო განყენებული არითმეტიკული ოპერაციების „გაუცნობიერებელი“ შემსრულებელი; მას ეს ოპერაციები მიზანდასხულად – კონკრეტული მრავალწევრის მნიშვნელობების გამოსათვლელად უნდა შეესრულებინა. *ჩვეულებრივი საანგარიშო მანქანისათვის* თუ საკმარისი იყო ერთი ან ორი ოპერანდის მიწოდება, რომელზეც კონკრეტული არითმეტიკული ოპერაცია უნდა შესრულებულიყო, *სხვაობით მანქანას* მრავალწევრის მნიშვნელობების გამოსათვლელად სჭირდებოდა სხვადასხვა ოპერანდზე არითმეტიკული ოპერაციების შესასრულებელ ბრძანებათა (ინსტრუქციათა) სპეციალურად ორგანიზებული მიმდევრობა, ანუ *პროგრამა*. ამ გაგებით *სხვაობითი მანქანა ერთპროგრამულ გამოთვლელ მანქანას* შეიძლება შეგვედარებინა, მაგრამ ასეთი შედარება არაკორექტული იქნებოდა: მას გარედან მიწოდებული პროგრამა კი არ უნდა შეესრულებინა, არამედ ამ პროგრამის შესაბამისად უნდა აგებულიყო მისი სტრუქტურა, ე. ი. ეს უკანასკნელი შეიძლებოდა წარმოგვედგინა როგორც „გაყინული“ კონკრეტულ პროგრამა. თანამედროვე ტერმინოლოგიის მიხედვით იგი რეალიზებული უნდა ყოფილიყო აპარატურულად და არა პროგრამულად.



ს. ბ. გ. ა. მ. ბ. გ. ა.

$\Delta_n^{\alpha}$	$\Delta_n^{\alpha} \mid N = n^4$	$\Delta_1$	$\Delta_2$	$\Delta_3$	$\Delta_4$
1	$1^4 = 1$	$16 - 1 = 15$	$65 - 15 = 50$	$110 - 50 = 60$	$84 - 60 = 24$
2	$2^4 = 16$	$81 - 16 = 65$	$175 - 65 = 110$	$194 - 110 = 84$	$108 - 84 = 24$
3	$3^4 = 81$	$256 - 81 = 175$	$369 - 175 = 194$	$302 - 194 = 108$	24
4	$4^4 = 256$	$625 - 256 = 369$	$671 - 369 = 302$	$108 + 24 = 132$	24
5	$5^4 = 625$	$1296 - 625 = 671$	$302 + 132 = 434$	$132 + 24 = 156$	24
6	$6^4 = 1296$	$671 + 434 = 1105$	$434 + 156 = 590$	$156 + 24 = 180$	24
7	$7^4 = 1296 + 1105 = 2401$	$1105 + 590 = 1695$	$590 + 180 = 770$	$180 + 24 = 204$	24
8	$8^4 = 2401 + 1695 = 4096$	...	...	...	...

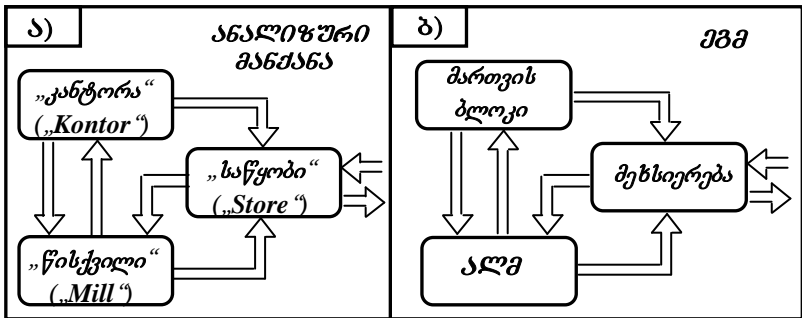
ს. ბ. 1.2 სხვაობათა სერების მათემატიკური რეკურსიული განმარტება

**1822** წელს **ბებიჯმა** ლილვებითა და კბილანათვლებით ააგო სხვაობითი მანქანის *მოდელი*. ბუნების შესახებ ცოდნის განვითარების ლონდონურმა სამეფო ორგანიზაციამ (*The Royal Society of London for the Improvement of Natural Knowledge*) მისი საქმიანობა „უმალლესი ხარისხის საზოგადოებრივი მხარდაჭერის ღირს“ საქმიანობად ჩათვალა და დაწვებული საქმიანობის ბოლომდე მისაყვანად **1823** წელს ბრიტანიის მთავრობამ მას საბოლოო ანგარიშით **1700** ფუნტი სტერლინგის ტოლი სუბსიდია გამოუყო.

მანქანაზე მუშაობის დასაწყისში **ბებიჯმა** ვერ გაიაზრა მის რეალიზაციასთან დაკავშირებული ყველა სირთულე: სამუშაო გამოყოფილი **3** წლის განმავლობაში ვერ დაასრულა და **9** წლის შემდეგ შეაჩერა მასზე მუშაობა. მანქანის უკვე დასრულებული ნაწილი შესანიშნავად მუშაობდა და გამოთვლებს მოსალოდნელზე უფრო მაღალი სიზუსტით ასრულებდა. ბებიჯმა ამ მანქანიდან ინფორმაციის გამოტანის ძალიან საინტერესო მოწყობილობაც შექმნა: იგი ინფორმაციას ფოლადის შტამპით ამოტვიფრავდა სპილენძის პატარა დაფაზე და ფაქტობრივად თანამედროვე პრინტერების წინასახეს – *მექანიკურ პრინტერს* წარმოადგენდა.

სხვაობითი მანქანის შექმნის დაფინანსების შეწყვეტის შემდეგ **ბებიჯმა**, გაიაზრა რა ერთი პროგრამის მარეალიზებელი აპარატურული გამომთვლელი მანქანის არასრულფასოვნება, შეუდგა **მრავალი პროგრამის** შემსრულებელი „*ანალიზური მანქანის*“ დაპროექტებას, თუმცა **1847-1849** წლებში ხელახლა დაუბრუნდა სხვაობით მანქანაზე მუშაობას, რომელსაც მან „*№2 ანალიზური მანქანა*“ (*Difference Engine №2*) უწოდა. ბებიჯის შრომებსა და რჩევებზე დაყრდნობით შვედმა გამოგონებელმა **გეორგ შუტცმა** (*Georg Scheutz 1785-1873*) **1854** წლიდან დაწყებული რამდენიმე სხვაობითი მანქანა ააგო და ერთ-ერთი მათგანი **1859** წელს ინგლისის მთავრობასაც მიჰყიდა. **1855** წელს **შუტცის სხვაობითმა მანქანამ** პარიზის მსოფლიო გამოფენაზე ოქროს მედალი დაიმსახურა. გარკვეული დროის შემდეგ მეორე შვედმა გამოგონებელმა **მარტინ ვიბერმა** (*Martin Wiberg, 1826-1905*) გააუმჯობესა **შუტცის** მანქანა და იგი ლოგარითმული ცხრილების გამოსათვლელად გამოიყენა.

სხვაობითი მანქანა უკვე არ იყო არითმეტიკული ოპერაციების შემსრულებელი მორიგი საანგარიშო მანქანა: იგი წარმოადგენდა **ვარკვეული პროგრამის აპარატურულად მარეალიზებელ** მოწყობილობას, რაც თვისობრივად განსხვავდება უბრალო კალკულატორისაგან. მასზე მუშაობის ლოგიკურ დასასრულამდე მიყვანამდე, როგორც ზემოთ აღვნიშნეთ, **ბებიჯმა** გაიაზრა ჯერ კიდევ არარეალიზებული მანქანის შეზღუდული შესაძლებლობა და გადაწყვიტა **შექმნა პროგრამული გამოთვლელი მანქანა**, რომელსაც მან **ანალიზური მანქანა** უწოდა. მისმა შექმნის იდეამ იმდენად გაიტაცა იგი, რომ მთელი დანარჩენი სიცოცხლე სწორედ მის რეალიზებას მიუძღვნა. უპირველეს ყოვლისა **ბებიჯმა** დაამუშავა ასეთი მანქანის სტრუქტურა. ამ მანქანის კონსტრუქციაში მან ჩართო „წისქვილად“ წოდებული არითმეტიკულ-ლოგიკური **ალმ** მოწყობილობა, „საწყობად“ წოდებული **მეხსიერება** და „კანტორად“ წოდებულ მარ-



ნახ. 1.3. ბებიჯის ანალიზური მანქანისა (ა) და ელექტრონული გამოთვლელი მანქანის (ბ) სქემები

**რთვის ბლოკი** (ნახ. 1.3,ა). თანამედროვე ტერმინებით ბებიჯისეული ტერმინების შეცვლის შემთხვევაში **ანალიზური მანქანის** ბლოკური სქემა ემთხვევა თანამედროვე ელექტრონული გამოთვლელი მანქანის (**ეეშ**-ის) ბლოკურ სქემას (ნახ. 1.3, ბ). ბებიჯი გახდა **კომპიუტერული შეტანა/გამოტანის იდეის** პიონერიც, რამდენადაც მისი პრინტერი და **ჟაკარდისეული** პერფობარათების დასტა საშუალებას საშუალებას იძლეოდა, გამოთვლელ მოწყობილობაში სრულიად ავტომატურად მომხდარიყო ინფორმაციის შეტანა და გამოტანა.

კომპიუტერულ მეცნიერებაში შეტანილი ბებიჯის წვლილის სიძლიერე, უპირველეს ყოვლისა, ის არის, რომ მან: **1)** იდეები სრულყოფილად ჩამოაყალიბა; **2)** დააპროექტა პერფორატორების მეშვეობით დაპროგრამებადი სისტემა; **3)** მანქანას მისცა სხვადასხვა ტიპის გამოთვლების უნარი; **4)** შეძლო მანქანა იმდენად მოქნილი გაეხადა, რამდენადაც ამის შესაძლებლობას იძლეოდა მის შესასვლელზე მიწოდებული ინსტრუქციები; სხვა სიტყვებით რომ ვთქვათ, „**ანალიზური მანქანის**“ მოქნილობას უზრუნველყოფდა **პროგრამული საშუალებები**; **5)** დაამუშავა პირველი მექანიკური პრინტერის კონსტრუქცია; **6)** სათავე დაუდო გამომთვლელ მანქანაში ინფორმაციის ავტომატურად შეტანისა და გამოტანის სისტემას.

„**ანალიზურ მანქანაში**“, გარდა ზემოთ აღნიშნულისა, რეალიზებული იყო **პირობითი გადასვლები**, გამოთვლების **განშტოებული ალგორითმები** და ორგანიზებული იყო ერთსა და იმავე ქვეპროგრამის მრავალჯერადად გამეორებისათვის საჭირო **ციკლები**.

ბებიჯმა იმდენად პროგრესული თეორიული მასალა დაამუშავა, რომ მან შეძლო დაიენტერესებინა და საკუთარი ჰიპოტეტური მანქანისათვის საჭირო პროგრამების შესადგენად მიეზიდა **ჯორჯ ბაირონის** ქალიშვილი, უდავო მათემატიკური ნიჭით დაჯილდოებული გრაფინია **ადა ლავლეისი (1815-1852)**, რომელიც გახდა მსოფლიოში **პირველი დამპროგრამებელი**.

**ჩარლზ ბებიჯმა**, სამწუხაროდ, ვერ შეძლო საკუთარი რევოლუციური იდეების უმეტესობის რეალური ხორცშესხმა. მეცნიერული მუშაობის მთელი პერიოდის განმავლობაში მას მუდამ თან სდევდა უსერიოზულესი პრობლემები.

**პირველი პრობლემა**, რომელიც ხელს უშლიდა **ჩარლზ ბებიჯს** სწრაფად მოეხდინა საკუთარი იდეების რეალიზება, იყო ის, რომ მის უსხარტეს გონებას არ შეეძლო ერთ ადგილზე გაჩერება და მორიგი ეტაპის დასრულების ლოდინი. დასამზადებელი კვანძის ნახაზების ოსტატებთან მიტანისთანავე **ბებიჯი** იწყებდა მასში შესწორებებისა და დამატებების შეტანას, განუწყვეტლივ ეძებდა მოწყობილობის გამართვებისა და გაუმჯობესების გზებს. ამ თვისების გამო ოსტატებს არ სიამოვნებდათ მასთან საქმის დაჭერა.

**მეორე პრობლემა** იყო გულფიცხოვა და კონფლიქტურობა. მას შეეძლო შეურაცყოფის ზღვარზე მდგარი გამოთქმები უხეშად პირში

მიხვალა არაკომპენტენტური მოპაექრისათვის. გასაგებია, რომ ასეთი ნატურისა და მკვეთრი საუბრების მოყვარულ მეცნიერს მუდამ ჰქონდა გართულებები არა მარტო მთავრობის წევრებთან, არამედ სასულიერო პირებთანაც.

1990-იანი წლების დასაწყისამდე ითვლებოდა, რომ **ჩარლზ ბე-ბიჯის** იდეებმა იმდენად გაუსწრო თავის დროს, რომ მის მიერ დაპროექტებული მანქანები პრინციპულად შეუძლებელი იყო აგებულიყო იმ ეპოქაში. ეს შეხედულება მხოლოდ **ჩარლზ ბებიჯის** დაბადების 200 წლისთავზე - 1991 წელს გაქარწყლდა: ლონდონის მუზეუმის თანამშრომლებმა ბებიჯისეული ნახაზების საფუძველზე და XIX საუკუნის ტექნოლოგიების გამოყენებით აღადგინეს 2,6-ტონიანი „№2 სხვაობით მანქანა“, ხოლო 2000 წელს - 3,5-ტონიანი **ბებიჯისეული პრინტერი**. ორივე მათგანი დღემდე შესანიშნავად მუშაობს, რამაც დაამტკიცა, რომ საჭირო დაინტერესების შემთხვევაში ისინი შესაძლებელი იყო XIX საუკუნეშივე აეგოთ.

### 1.3. პირველი გამოთვლილი მანქანები

პირველ პარაგრაფში განხილულისაანგარიშო მანქანებს ცალკეული არითმეტიკული ოპერაციების შესრულება ხელეწიფებოდა. ბე-ბიჯის „**სხვაობითი მანქანა**“ გამიზნული იყო არა ცალკეული არითმეტიკული ოპერაციების, არამედ ამ ოპერაციების წინასწარ შერჩეული მიმდევრობის შესასრულებლად. ცალკე აღებული ერთი ბრძანება იმდენად ტრივიალურია, რომ არ შეიძლება რეალურ პროგრამად განვიხილოთ, მაგრამ ბრძანებათა გარკვეული ერთობლიობა უკვე პროგრამას წარმოადგენს. ამგვარად, „**სხვაობითი მანქანა**“ სასურველი სამუშაოს პროგრამულად მარეალიზებელ მანქანად წარმოგვიდგება, ოღონდ ეს პროგრამა მას გარედან კი არ ეწოდებოდა, არამედ ამ პროგრამის შესაბამისად იყო აგებული მისი სტრუქტურა. ამდენად, გარკვეული დაშვებით, იგი **აპარატურულად რეალიზებულ მანქანის** პირველსახედ შეიძლება განვიხილოთ. სხვა პროგრამის შესრულებისათვის აუცილებელი იყო მისი კონსტრუქციის დაშლა და ახალი კონსტრუქციის აგება, რაც ახალი მანქანის აგების ტოლფასია და დიდ დანახარჯებთანაა დაკავშირებული. **ჩარლზ ბებიჯი** არ

დააკმაყოფილა ასეთი მანქანის აგების იდეამ და შეეცადა ისეთი მანქანა აეგო, რომლიც კონსტრუქციის შეუცვლელად შეასრულებდა რამდენიმე პროგრამას. მან გაიაზრა, რომ ეს პროგრამები მანქანას გარედან უნდა მიწოდებოდა და შენახულიყო მის საწყობშიანუ მენსიერებაში (იხ. ნახ. 1.2). მაშასადამე, მას დაებადა აპარატურულად რეალიზებული მანქანიდან პროგრამულად რეალიზებულ მანქანაზე გადასვლის იდეა და შეუდგა მის განხორციელებას. ამ პერიოდში არსებული საელემენტო ბაზის შეზღუდული შესაძლებლობების გამო მან ვერ შეძლო დაწყებული სამუშაოს ლოგიკურ დასასრულამდე მიყვანა, მაგრამ დატოვა საკმაო მოცულობის თეორიული და პრაქტიკული გადაწყვეტები, რომელთა გამოყენება მომავალ პიონერებს გაუადვილებდა საქმიანობას.

**ჩარლზ ბებიჯის** მიერ დაწყებულ საქმიანობას **XX** საუკუნის **30**-იან წლებში გამოუჩნდა მრავალი ღირსეული გამგრძელებელი, რომელთა შორის უნდა აღვნიშნოთ ბულგარელი წარმოშობის ამერიკელი ფიზიკოსი, მათემატიკოსი და ელექტროინჟინერი **ჯონ ათანასოვი (1903-1995)**, გერმანელი ინჟინერი **კონრად ცუზე (1910-1955)**, ინგლისელი მათემატიკოსი, ლოგიკოსი და კრიპტოლოგი **ალან ტიურინგი (1912-1954)**, ამერიკელი ფიზიკოსი **ჰოვარდ ეიკენი (1900-1973)** და სხვები. მოკლედ განვიხილოთ გამოთვლითი ტექნიკის განვითარებაში მათ მიერ შეტანილი წვლილი.

■ დაიწყებთ **აიოვას** შტატის (**აშშ**) სახელმწიფო კოლეჯის პროფესორ **ჯონ ათანასოვი**. თავდაპირველად იგი ცდილობდა განტოლებათა სისტემის ამოსახსნელად **IBM** ფირმის მიერ შექმნილი კალკულატორის მოდიფიცირებას, მაგრამ მალე ამაზე ხელი აიღო და გადაწყვიტა კომპიუტერის შექმნაზე დაეწყო მუშაობა. კოლეჯის საინჟინრო ფაკულტეტის დეკანს მან სთხოვა ამ საქმეში ჩასართველად რეკომენდაცია მიეცა ფაკულტეტისგამომშვებული სპეციალისტისათვის, რომელმაც კარგად იცოდა ელექტრონიკა. დეკანმა მას შესთავაზა **კლიფორდ ედვარდ ბერის (1918-1963)** კანდიდატურა, რომელიც გახდა ათანასოვის ასპირანტი. მათ დაიწვეს **ათანასოვის** მიერ ჩაფიქრებულ კომპიუტერზე მუშაობა, რომელსაც **ათანასოვმა ABC** უწოდა (**A**tanas-off **B**erry **C**omputer). იგი არ იყო დაპროგრამებადი და განკუთვნილი იყო მხოლოდ **წრფივი განტო-**

*ლებების* ამოსახსნელად. **1942** წელს **ABC** უკვე წარმატებით აგებულიცა და ტესტირებულიც იყო. მასში არსებულ მბრუნავ დოლზე განთავსებული იყო ქალაქის **კონდენსატორებიანი** დამხსომებული მოწყობილობა. იგი საკმაოდ არასაიმედოდ, მაგრამ მაინც ფუნქციონირებდა. თანამედროვე კომპიუტერებში **ოპერატიულ მეხსიერებად**სწორედ **ათანასოვის** მიერ შემოთავაზებული, ოღონდ სათანადოდ მოდერნიზებული, **კონდენსატორული მეხსიერება** გამოყენებული. კომპიუტერზე სრულყოფაზე მუშაობა შეწყდა იმის გამო, რომ ათანასოვმა დატოვა აიოვას შტატის კოლეჯი.

**ABC**-ში პირველად გამოჩნდა თანამედროვე კომპიუტერიისათვის დამახასიათებელი ისეთი ელემენტები, როგორცაა, მაგალითად, ორობითი არითმეტიკა და ტრიეგერები; თანამედროვე კომპიუტერისაგან განსხვავებით იგი იყო არა უნივერსალური, არამედ სპეციალიზებული კომპიუტერი, რომელშიც შეუძლებელი იყო გამოთვლების ცვალებადობა, რადგან მის მეხსიერებაში არ შეინახებოდა კომპიუტერული პროგრამა.

კომპიუტერების შექმნის საქმეში **ათანასოვისა** და **ბერის** მიერ გაწეული ღვაწლი დიდი ხნის განმავლობაში სათანადოდ არ იყო შეფასებული. **ჯონ მოუჩლის** ჯგუფის მიერ **1945** წელს შექმნილი გამოთვლელი მანქანა **ENIAC** (მასზე ქვემოთ ვისაუბრებთ) ჩაითვალა პირველ კომპიუტერად და **მოუჩლის** სახელზე გაიკა სათანადო პატენტი. **1973** წელს **აშშ**-ს რაიონულმა ფედერალურმა სასამართლომ ეს პატენტი გააუქმა და კომპიუტერის შექმნის პირველობა **ჯ. ათანასოვს** მიაკუთვნა, ე.ი. პირველ კომპიუტერად **ABC** კომპიუტერი იქნა აღიარებული.

**კონრად ცუზემ** არ იცნობდა **ჩარლზ ბებიჯის** შრომებს, ამიტომ იგი **ჩარლზ ბებიჯის** მიერ დაწეებულ საქმეს ვერ გააგრძელებდა. მათ ერთმანეთთან ის აკავშირებს, რომ ორივეს დამოუკიდებლად დაებადათ კომპიუტერის შექმნის იდეა. **ჩარლზ ბებიჯმა** თუ დაგვიტოვა ამ იდეის რეალიზებისათვის აუცილებელი თეორიულ-პრაქტიკული მასალა, **კონრად ცუზემ** შეძლო ერთ-ერთ პირველთაგანს შესხა ხორცი აღნიშნული იდეისათვის.

**კონრად ცუზემ 1935** წელს დაამთავრა ბერლინის უმაღლესი ტექნიკური სკოლა, რომელსაც დღეს **ბერლინის ტექნიკური უნივერსიტეტი** ჰქვია და დაიწყო დაპროგრამებადი გამომთვლელი მანქანის შექმნაზე მუშაობა. საკუთარი ხარჯებით **1938** წელს მან ააგო **Z1** სახელწოდების მანქანა. ის წარმოადგენდა ელექტრულ ამძრავიან და კლავიატურის საშუალებით შეზღუდულად დაპროგრამებად ორობით ელექტრონულ გამომთვლელს. თვლის ათობითი სისტემის გამოყენებით წარმოდგენილი შედეგი აისახებოდა მილაკურ პანელზე. **Z1** წარმოადგენდა არასაიმედოდ მომუშავე **საექსპერიმენტო მოდელს** და პრაქტიკული ამოცანების გადასაწყვეტად არ იყო განკუთვნილი.

**1940** წელს აეროდინამიკის კვლევითმა ინსტიტუტმა გადაწყვიტა **ცუზე** ჩაერთო მართული რაკეტების შექმნაში და მას აღმოუჩინა ფინანსური მხარდაჭერა, რისი შემწეობითაც **ცუზემ** ააგო **Z2** ვერსიის მანქანა. **Z1**-ისაგან განსხვავებით იგი ინსტრუქციებს კითხულობდა **35**-მილიმეტრიანი კინოფირისგან და წარმოადგენდა **სადემონსტრაციო მოდელს**. მის საფუძველზე **ცუზემ 1941** წელს შექმნა სრულყოფილი **Z3** მოდელი, რომელსაც დღეს ბევრი რეალურად მოქმედ **პირველ რელეურ მანქანად** მიიჩნევენ. იგი შეიცავდა დაახლოებით **2600** რელეს, რომელთაგანაც **1400** რელეს საშუალებით აგებული იყო **მეხსიერების მოდული**, **600** რელესაგან – **ართიმეტიკული მოდული**, ხოლო დანარჩენი რელეებისაგან – **მართვის სქემები**. სამივე - **Z1**, **Z2** და **Z3** მანქანა განადგურდა **1944** წელს ბერლინის დაბომბვის დროს.

**1950** წლის სექტემბერში **ცუზემ** ააგო **Z4** მანქანა. იმ პერიოდში იგი წარმოადგენდა კონტინენტურ ევროპაში მომუშავე პირველ მოქმედ ელექტრონულ გამომთვლელ მანქანას, რომელიც **ბაზარზე გაიყიდა**. ამ მხრივ მან **5** თვით დაასწრო **ჰოვარდ ეიკენის** მიერ ამერიკაში აგებულ რელეურ მანქანა **Mark I**-ს და **10** თვით **ჯონ ეკერტისა** და **ჯონ მოკლის** მიერ ასევე ამერიკაში აგებულ ელექტრონული-მილაკურ მანქანა **UNIVAC**-ს.

■ კომპიუტერული მეცნიერების სფეროში გამოქვეყნებულ პირველ სამეცნიერო ნაშრომად ითვლება ინგლისელი მათემატიკოსის, ლოგიკოსისა და კრიპტოგრაფის **ალან ტიურინგის** მიერ **1936**



წლის 12 ნოემბერს გამოქვეყნებული ცნობილი სამეცნიერო სტატია „*On Computable Numbers, with an Application to the Entscheidungsproblem*“, რომელშიც მან შემოგვთავაზა ჰიპოთეტურ უნივერსალური გარდამქმნელი მანქანის კონცეფცია; აღნიშნული მანქანას მოგვიანებით შექმნილის საპატივცემლოდ „*ტიურინგის მანქანა*“ უწოდეს.

„*ტიურინგის მანქანა*“ შეიძლება ჩაითვალოს საერთო დანიშნულების კომპიუტერად. *საერთო დანიშნულების კომპიუტერი* (*general purpose computer*) ეწოდება კომპიუტერს, რომელსაც შეუძლია გადაწყვეტილოს ნებისმიერი სახის ამოცანა, თუ: *ა)* იგი წარმოდგენილია მენსიერებაში შესანახად დასაშვები ზომის მქონე პროგრამის სახით; *ბ)* მისი გადაწყვეტის მოთხოვნილი სინქარე არ აღემატება კომპიუტერის მუშაობის საჩქარეს, და *გ)* ამის საშუალებას იძლევა კომპიუტერის მოწყობილობათა საიმედოობა.

*ტიურინგმა* თავისი აბსტრაქტული მანქანის მეშვეობით გვიჩვენა, რომ კომპიუტერების მეშვეობით შეგვიძლია გადავწყვიტოთ ნებისმიერი პრობლემა, თუ, *ჯერ ერთი*, ავაგებთ სათანადო ტექნიკური მჩვენებლების მქონე კომპიუტერს, და, *მეორეც* - მოვახერხებთ პრობლემის ალგორითმიზებას.

მეორე მსოფლიოს დროს *ალან ტიურინგი* კოდებისა და შიფრების სახელმწიფო სკოლაში ხელმძღვანელობდა ჯგუფს, რომელიც პასუხისმგებელი იყო გერმანიის სამხედრო-საზღვაო ფლოტის შეტყობინებათა კრიპტოანალიზზე.

*ნისლიან ალბიონში* (*ბრიტანეთში*) პირველი დაპროგრამებადი გამოთვლელი მანქანის შექმნის ისტორია დიდი ხნის განმავლობაში ნისლით იყო მოცული. მეორე მსოფლიო ომის დასაწყისში გერმანიის წყალქვეშა გემები სერიოზულ ზიანს აყენებდა ბრიტანეთის ფლოტს. გერმანელი გენერლები თავის წყალქვეშა გემებს ბრიტანეთის გემებზე თავდასხმის ბრძანებებს იმ დროს გაგრცელებული მორხეს აპარატის ნაცვლად რადიოკავშირის საშუალებით უგზავნიდნენ. საქმეს არ შეელოდა ის გარემოება, რომ ინგლისის სპეცსამსახურებიც იღებდა ამ ბრძანებებს. ამას განაპირობებდა ის, რომ ბრძანებები დაშიფრული იყო დილეთანტი გამოძიებლის, აშშ-ის ყოფილი პრეზიდენტის *თომას ჯეფერსონის* მიერ დაპროექტებული

ხელსაწყოს *ENIGMA*-ს საშუალებით და მის გასაშიფრად დიდი რაოდენობის გამოთვლების ჩატარება იყო საჭირო. ამ გამოთვლების შესრულებას რამდენიმე კვირა სჭირდებოდა და ამ დროის განმავლობაში ბრძანებები უკვე შესრულებული იყო და მათი შინაარსის ცოდნას არავითარი მნიშვნელობა არ ჰქონდა.

ბრიტანეთის მთავრობამ დააარსა საიდუმლო ლაბორატორია, რომლის მიზანი იყო შეექმნა ზემოთ აღნიშნული გამოთვლების სწრაფად შემსრულებელი საიდუმლო გამოძიებელი მანქანა *COLOSSUS*. ამ ლაბორატორიაში შეყვანილი იქნა *ალან ტიურინგი*. **1943** წელს აღნიშნულმა ჯგუფმა ააგო ასეთი მანქანა. იგი შედგებოდა **1500** ელექტრონული მილაკისაგან და იმ პერიოდში ყველაზე დიდ გამოძიებელ მანქანას წარმოადგენდა. მისი გამოყენების შემთხვევაში მიღებული შეტყობინებების გასაშიფრად რამდენიმე კვირის ნაცვლად ორი საათი იყო საჭირო. მოდერნიზებული *COLOSSUS Mark II* უკვე **2500** ელექტრონულ მილაკს შეიცავდა და ელექტრონული გამოძიებელი მანქანების ისტორიაში პირველ დაპროგრამებად კომპიუტერსად ითვლება. ომის დასასრულისათვის ინგლისში უკვე ათი *COLOSSUS* ფუნქციონირებდა. ისინი **30** წლის განმავლობაში გასაიდუმლებული იყო და ამიტომ ისინი კომპიუტერული ტექნიკის განვითარების საქმეში ბაზად არ იქნა გამოყენებული. მის შექმნაში შეტანილი დამსახურებისათვის **1945** წელს *ალან ტიურინგი* დაჯილდოვდა ბრიტანეთის იმპერიის ორდენით. **1966** წლიდან გამოთვლითი ტექნიკის ასოციაცია ყოველწლიურად გასცემს *ტიურინგის პრემიას (Turing Award)*, რომელიც ინფორმატიკაში ყველაზე პრესტიჟული პრემიაა.

■ **კოზარდ ვიკენს XX** საუკუნის **30**-იან წლებში ჰარვარდის უნივერსიტეტში (**აშშ**) ფიზიკის სფეროში მიიღო ფილოსოფიის დოქტორის ხარისხზე მუშაობის დროს მოუხდა ისეთი დიფერენციალური განტოლებების გამოყენება, რომელთა ამოხსნა მხოლოდ რიცხვითი მეთოდებით შეიძლებოდა. მათ ამოხსნაზე მუშაობისას *ვიკენს* მსგავსი რუტინული მათემატიკური გამოთვლების შემსრულებელი ელექტრომექანიკური მოწყობილობის აგების იდეა დაებადა. სადოქტორო დისერტაციის დაცვის (**1939** წ.) ამ იდეის განსახორციელებლად მან გულმოდგინედ შეისწავლა *ჩარლზ ბებიჯის* შრომები, რო-

მლებშიც იგი აღაფრთოვანა *ბებიჯის „სხვაობით მანქანამ“*. მისი ხელმძღვანელობით მომუშავე ინჟინერთა ჯგუფმა **IBM**-თან დადებული კონტრაქტის შესაბამისად **1941** წელს ელექტრომაგნიტური რელეებით ააგო „*Automatic Sequence Controlled Calculator*“ („მიმდევრობით მართვადი ავტომატური კალკულატორი“) სახელწოდების დაპროგრამებადი გამომთვლელი მანქანა, რომელსაც **Mark I** უწოდა. სავალდებულო ტესტირების წარმატებით გავლის შემდეგ **1946** წელს იგი ჰარვარდის უნივერსიტეტში გადაიტანეს. მოცემული გამომთვლელი მანქანა შეიცავდა **765000** დეტალს (ელექტრომაგნიტურ რელეებს, გადამრთველებს და ა.შ.), იწონიდა დაახლოებით **4,5** ტონას, მისი სიგრძე იყო **17** მეტრი, სიმაღლე - **2,5** მეტრი, ხოლო მასში გამოყენებული სადენების სიგრძე თითქმის **800** კილო-მეტრს აღწევდა. ძირითადი გამომთვლელი მოდულები სინქრონიზდებოდა **15** მეტრიანი ლილვით, რომელსაც აბრუნებდა **5** ცხენის ძა-ლიანი (**4** კილოვატიანი) ელექტრული ძრავა. ფაქტობრივად **Mark I** გაუმჯობესებულ *არითმომეტრს* წარმოადგენდა, რომლის ჩვეულებრივი სახელურიანი საანგარიშო მანქანებით აღჭურვილი დაახლოებით **20** გამომთვლელის შრომას ცვლიდა, მაგრამ ვინაიდან დაპროგრამების უნარი ჰქონდა, ამიტომ ხშირად მას *რეალურად მომქმედ ელექტრომექანიკურ გამომთვლელ მანქანასაც* უწოდებენ.

**IBM**-თან დავის მიუხედავად ეიკენმა გააგრძელა საკუთარ გამომთვლელი მანქანებზე მუშაობა. **1947** წელს დაასრულა „*ჰარვარდული Mark II*“, რომელსაც მოჰყვა „*ჰარვარდული Mark III*“ (**1949**) და „*ჰარვარდული Mark IV*“ (**1952**). *მარკ III*-ში უკვე გამოყენებული იყო ზოგიერთი ელექტრონული კომპონენტები, ხოლო **Mark IV** უკვე მთლიანად ელექტრონულ მანქანას წარმოადგენდა.

#### 1.4. გამომთვლელი მანქანების აგების კლასიკური პრინციპები

გამომთვლელი მანქანების აგების კლასიკური პრინციპები *ა. ბი-ორკსმა (1915-2008)*, *პ. ვოლდსტაინმა (1913-2004)* და *ჯონ ფონ ნიუმანმა (1903-1957)* ჩამოაყალიბეს **1946** გამოქვეყნებულ სტატი-

აში, *„Preliminary Discussion of the Logical Design of an Electronic Computing Instrument“* („ელექტრონული გამოთვლელი მოწყობილობის ლოგიკური კონსტრუირების წინასწარი განხილვა“). ჩამოთვლილ ავტორებს შორის ყველაზე ცნობილ მეცნიერს წარმოადგენდა **ჯონ ფონ ნეიმანი**, რის გამოც აღნიშნულ პრინციპებს ხშირად **ფონ ნეიმანისეულ პრინციპებსაც** უწოდებენ.

ზემოთ აღნიშნული სტატიაში გამოთვლითი მანქანების აგების პრინციპები დანომრილი ზუსტი პოსტულატების სახით არ არის ჩამოთვლილი, ამიტომ სხვადასხვა სახელმძღვანელოში ისინი განსხვავებული თანამიმდევრობით სხვადასხვანაირად არის ფორმულირებული. ჩვენ მათ შემოგვთავაზებთ ჩვენებური ინტერპრეტაციით.

სტატიის დასაწყისში განსაზღვრულია გამოთვლელი მანქანის ძირითადი კომპონენტები, მათი დანიშნულება და შედგენილობა; შემდეგ განხილულია ორობითი კოდირების, მეხსიერების დამისამართებისა და იერარქიული ორგანიზების, შენახვადი მეხსიერების აგების და პროგრამული მართვის ძირითადი პრინციპები. ჩვენ ზემოთ ჩამოთვლილი თანამიმდევრობით განვიხილავთ თითოეულ მათგანს.

■ **მანქანის ძირითადი კომპონენტები.** ასეთი სახელწოდების პირველ განყოფილებაში **ფონ ნეიმანმა** თანაავტორებთან ერთად განსაზღვრა და დაასაბუთა ელექტრონული გამოთვლელი მანქანის შემადგენლობა. მათი აზრით უნივერსალურ გამოთვლელ მანქანას აუცილებლად უნდა ჰქონოდა ისეთი ძირითადი კვანძები, როგორცაა არითმეტიკულ-ლოგიკური კვანძი, მეხსიერება, მართვისა და ოპერატორთან კავშირის კვანძი; მანქანა უნდა ყოფილიყო ავტომატური, ე. ი. ამუშავების შემდეგ მისი ფუნქციონირება არ უნდა ყოფილიყო დამოკიდებული ოპერატორზე.

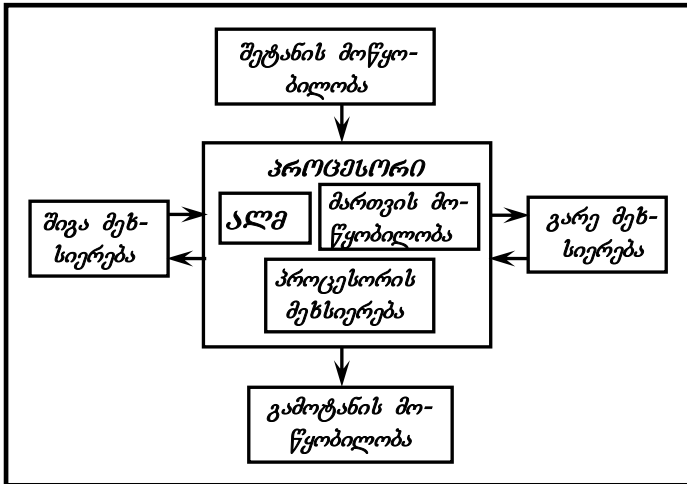
მაშასადამე, გამოთვლელი მანქანა უნდა შედგებოდეს ზუსტად განსაზღვრული ფუნქციების შემსრულებელი რამდენიმე კვანძისაგან. ეს კვანძები არსებობს თანამედროვე კომპიუტერებშიც:

■ **არითმეტიკულ-ლოგიკური მოწყობილობა (ალმ)**, რომელშიც ხდება მონაცემების დამუშავება;

■ **მართვის მოწყობილობა**, რომელიც უზრუნველყოფს პროგრამის შესრულებასა და მანქანის ცალკეული კვანძების კოორდინირებულ მუშაობას. დღეს **ალმ** და მართვის **მოწყობილობა მიკროპროცესორის სახით** მზადდება. მაღალი სწრაფმოქმედების უზრუნველსა-

ყოფად მასში მცირე მოცულობის ზესწრაფი მეხსიერებებიცაა გათვალისწინებული, რომეელთა ერთობლიობას პირობითად **პროცესორული მეხსიერება** ვუწოდოთ;

■ **მეხსიერება** განკუთვნილია პროგრამებისა და მონაცემების შესანახად. იგი შეიძლება დავეოთ **შიგა** და **გარე მეხსიერებად**. **შიგა მეხსიერება**ში მონაცემები მხოლოდ მათი დამუშავების პერიოდის განმავლობაში, ე. ი. დროებით შეინახება, ხოლო გარე მეხსიერება განკუთვნილია დამუშავების სეანსებს შორის მონაცემების ხანგრძლივად შენახვისათვის. ზემოთ ფორმულირებული სპეციფიკის შესაბამისად შიგა მეხსიერებას ხშირად **ოპერატიულ**, ხოლო გარე მეხსიერებას – **ხანგრძლივად შენახვის მეხსიერებასაც** უწოდებენ.



ნახ.1.4. კლასიკური ვარიანტის გამოძველელი მანქანის ბლოკური სქემა

■ **შეტანის მოწყობილობა** კომპიუტერში შესატან მონაცემებს კომპიუტერის მიერ ადვილად აღსაქმელ ფორმას აძლევს და ისე აწოდებს მათ კომპიუტერს;

■ **გამოტანის მოწყობილობას** მანქანის მუშაობის შედეგები ადამიანის (მომხმარებლის) მიერ ადვილად აღქმადი ფორმით გამოაქვს კომპიუტერიდან.

კლასიკურ გამოთვლელ მანქანაში ყველა ეს მოწყობილობა პროცესორის მეშვეობით ურთიერთზემოქმედებს ერთმანეთზე (ნახ. 1.4).

**■ ორობითი კოდირების პრინციპი.** ორობითი ინფორმაციის შენახავი მოწყობილობები და მისი გადაშუშავების მეთოდები ყველაზე მარტივი და იაფია.

ადრეულ მანქანებში მხოლოდ ციფრული მონაცემები გამოიყენებოდა. დღეისათვის გამოთვლელი მანქანებით სხვა სახის (ტექსტურ, გრაფიკულ, აუდიო და ვიდეო) ინფორმაციებსაც ამუშავებენ, მაგრამ ამანაც ვერ გააუქმა ორობითი კოდირების პრინციპი. რეალურ დროში ციფრული სიგნალების დამამუშავებელი სასიგნალო პროცესორებიც (*Digital Signal Processor - DSP*) ორობითი სახით წარმოდგენილ მონაცემებს იყენებს. ისტორიიდან ცნობილია *Семьюн*-ის ტიპის ელექტრონულ გამოთვლელ მანქანაში რუსი მეცნიერის **ნ. პ. ბრუსენცოვის** მიერ თვლის სამობითი სისტემის წარმატებითი გამოყენების ფაქტიც, მაგრამ ამ კომპიუტერმა გავლენა ვერ მოახდინა გამოთვლითი ტექნიკის ევოლუციაზე. ეს, უპირველეს ყოვლისა, განაპირობა ნახევარგამტერული ტექნოლოგიის საფუძველზე სამობითი კომპიუტერისათვის საჭირო ელემენტების დამზადებისას წარმოშობილმა სერიოზულმა პრობლემებმა. მათი გადაწყვეტა ჯერ-ჯერობით ვერ ხერხდება, მაშინ როდესაც ორობითი კომპიუტერებისათვის მსგავსი მოწყობილობების მასობრივი წარმოება გახდა შესაძლებელი. მიუხედავად ამისა, დაპროგრამების სფეროს უდიდესი ავტორიტეტი **დონალდ ერვინ კნუტი** ამტკიცებს, რომ სამობითი ლოგიკა ორობით ლოგიკაზე გაცილებით ელევანტური და ეფექტურია და კაცობრიობა მომავალში, ალბათ, ისევ დაუბრუნდება სამობითი კომპიუტერების დამუშავებას.

დასასრულს უნდა შევნიშნოთ, რომ ორობითი კოდირების პრინციპმა მნიშვნელოვნად გააფართოვა გამოთვლელ მანქანაში გამოყენებადი ხელსაწყოებისა და მოვლენების ნომენკლატურა. მართლაც, ორობითი სისტემის დროს გვაქვს მხოლოდ ორი ციფრი **0** და **1**, ამიტომ მათი წარმოდგენისათვის შეგვიძლია გამოვიყენოთ ორი სტაბილური მდგომარეობის მქონე ნებისმიერი სისტემა. მაგალითად, **ტრიოდი** (ღია და დახურული მდგომარეობა), **ტრივერი** (ორი მდგრადი მდგომარეობა), **ფერომაგნიტური ზედაპირის** უბანი

(დამაგნიტებული ან განმაგნიტებული მდგომარეობა), *იმპულსური სქემა* (ელექტრული იმპულსის არსებობა და არარსებობა) და ა.შ. თვლის ორობითი სისტემის მიხედვით აგებული ლოგიკური სქემებისათვის შეგვიძლია გამოვიყენოთ ბულის ალგებრის მათემატიკური აპარატი. მაშასადამე, თვლის ორობითი სისტემა ჯერჯერობით არსებითად ამარტივებს გამოთვლითი მანქანის ტექნიკურ კონსტრუქციას, რის შედეგადაც მცირდება მისი ღირებულება და იზრდება სამელოობა.

**მეხსიერების დამისამართების პრინციპები.** მანქანის ოპერატიული მეხსიერება შედგება ცალკეული ბიტებისაგან. ჩასაწერად ან წასაკითხად მეზობელი ბიტების ჯგუფები საკუთარი მისამართის (ნომრის) მქონე *მეხსიერების უჯრედში* უნდა ჩაიწეროს. უჯრედების დამისამართება იწყება ნულიდან. მეხსიერების უჯრედის მისამართებად გამოიყენება რიცხვები, რომლებითაც ისინი ინომრება. მიღებულია, რომ დანომვრა დაიწყოს ნულიდან.

უჯრედში ჩაწერილ მონაცემს ამ *უჯრედის შიგთავსი* ეწოდება და მას აქვს ორობითი რიცხვის სახე. შიგთავსი შეიძლება მხოლოდ მთლიანად იქნეს წაკითხული, ე. ი. შეუძლებელია ამ მონაცემის შესაბამისი ორობითი რიცხვის ცალკეული ნაწილის დამოუკიდებლად წაკითხვა.

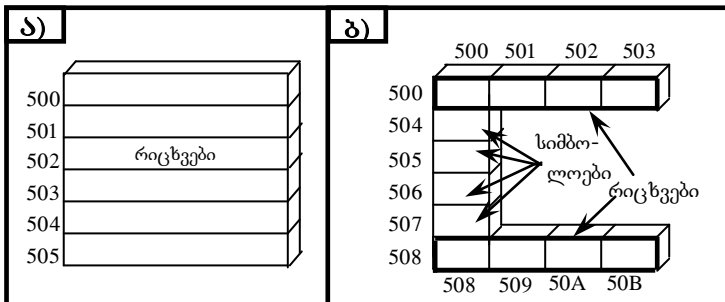
მეხსიერებაში საჭირო უჯრედების პოვნისათვის მისამართებად წოდებული რიცხვების გამოყენება აბსოლუტურად ბუნებრივად გამოიყურება: კომპიუტერში ყველაფერი რიცხვებით კოდირდება, ამიტომ ამ ფუნდამენტური წესიდან ვერც უჯრედების მისამართები გამოირიცხება. მეზობელი უჯრედების ნომრები ერთმანეთისაგან ერთით თუ იქნება განსხვავებული, მაშინ ისინი მიმდევრობით ადვილად დამუშავდება.

სხვადასხვა თაობის კომპიუტერების მეხსიერებათა უჯრედების თანრიგიანობა (უჯრედში არსებული ბიტების რაოდენობა) ერთმანეთისაგან განსხვავდება. თავდაპირველად მანქანები მხოლოდ მათემატიკური გამოთვლებისათვის გამოიყენებოდა. ამ დროს სასურველი იყო რიცხვები რაც შეიძლება ზუსტად ყოფილიყო წარმოდგენილი, ამიტომ ადრეულ მანქანებში ოპერატიული დამხსომებელი მოწყობილობის უჯრედები გრძელი იყო. გარდა ამისა, მანქანას მეხსიერებაში პროგრამის ბრძანებებიც უნდა შეენახა; იმ დროს, როგორც წე-

სი, რიცხვითი უჯრედის ზომა ემთხვეოდა ბრძანების ზომას, რაც მნიშვნელოვნად ამარტივებდა მეხსიერების მოწყობილობას.

გამომთვლელი მანქანების *მეორე თაობიდან მესამე თაობაზე* გადასვლის პერიოდში გამომთვლელ მანქანებზე სიმბოლური ინფორმაციის დამუშავებაც დაიწყო, რამაც სერიოზულად გაართულა საქმე: არსებულ რიცხვით უჯრედში **4-5** სიმბოლო ეტეოდა, რაც გამორიცხავდა ერთ სიმბოლოსთან დამოუკიდებლად შეღწევას. ინჟინრებმა პრობლემის უმარტივესი გადაწყვეტა აირჩიეს – უჯრედის ზომა ისე შეამცირეს, რომ თითოეულ სიმბოლოსთან განცალკევებულად მიმართვა ყოფილიყო შესაძლებელი. სიმბოლოს კოდირებისათვის გამოყენებული ბიტების ერთობლიობას **ბაიტი** უწოდეს. მაშასადამე **ბაიტი** არის ერთი სიმბოლოს კოდირებისათვის გამოსაყენებელი ბიტების ჯგუფი. სხვადასხვა შეხედულების მიუხედავად საბოლოოდ შეთანხმდნენ, რომ ერთ ბაიტში **8** ბიტი გააერთიანებინათ, რის გამოც ხშირად **ბაიტს** რვა ბიტის ერთობლიობასაც უწოდებენ. რვაბიტურ უჯრედზე დაფუძნებულმა **ბაიტურმა მეხსიერებამ** სპეციალისტების საყოველთაო მოწონება დაიმსახურა და იგი დღემდე გამოიყენება კომპიუტერულ ტექნიკაში (აღვნიშნავთ, რომ იყო პერიოდი, როდესაც სიმბოლოს კოდირებისათვის ხუთბიტურ რიცხვებსაც იყენებდნენ და მაშინ ბაიტად სწორედ ხუთბიტური რიცხვი ითვლებოდა).

მეხსიერების «მოკლე» უჯრედებზე გადასვლის შედეგად საჭირო გახდა რიცხვებს საკუთარი მისამართებიანი რამდენიმე უჯრედი (ბაიტი) დაეკავებინა. **1.5ა** ნახაზზე ნაჩვენებია თავდაპირველი, ხოლო **1.5ბ** ნახაზზე – თანამედროვე გამომთვლელი მანქანების მეხსიერების უჯრედების ორგანიზაცია,



**ნახ. 1.5**ადრეული (ა) და თანამედროვე (ბ) გამომთვლელი მანქანების მეხსიერების უჯრედების ორგანიზაცია



**1.5,ა** ნახაზზე რიცხვებს თითო-თითო უჯრედი უკავია, ამასთანავე ამ უჯრედების ნომრები ერთმანეთისაგან **1**-ით განსხვავდება. **1.5, ბ** ნახაზზე ნაჩვენებია ორი **32**-ბიტური რიცხვები, რომლებიც შენახულია **500-303** და **508-20B** ბაიტებში (მისამართები **16**-ობით სისტემაშია წარმოდგენილი). მიღებული წესის მიხედვით მისამართად მიღებულია უმცირესი მისამართები; ასე რომ, მოცემულ შემთხვევაში მისამართებს წარმოადგენს რიცხვები **500** და **508**. გარდა ამისა, **1.5, ბ** ნახაზზე წარმოდგენილ **32**-თანრიგიან რიცხვებს შორის არსებულ **504-507** ბაიტებში განთავსებულია ერთბაიტიური სიმბოლოები. შევნიშნავთ, რომ თანამედროვე კომპიუტერებს მეხსიერებასთან ყოველი მიმართვის დროს მისგან რვაბდე მეზობელი ბაიტიური უჯრედის შიგთავსების ამოღება შეუძლია.

განსაკუთრებით მნიშვნელოვანია ის, რომ ნებისმიერი თანამიმდევრობით შეიძლება უჯრედებიდან ინფორმაციის წაკითხვა და მათში ინფორმაციის ჩაწერა. რადგან ზემოთ აღნიშნული ოპერაციების ჩატარების თანამიმდევრობა *თავისუფლად* (რაიძე შეზღუდვის გარეშე) შეიძლება ამოვირჩიოთ, ამიტომ ამგვარად ორგანიზებულ მეხსიერებას *თავისუფლად შელწევად მეხსიერება* (ინგლისურად – *Random Access Memory* ანუ *RAM*) ეწოდება. მისგან განსხვავდება მაგნიტური ლენტისანი მეხსიერება, რომლიდანაც მონაცემები ამოღება მხოლოდ მიმდევრობითად და არა თავისუფლად წაკითხვის გზითაა შესაძლებელი; ამიტომ მას ეწოდება არა თავისუფლად შელწევადი, არამედ *მიმდევრობითად შელწევადი მეხსიერება*.

ხშირად ტერმინს „*თავისუფლად შელწევად მეხსიერებას*“ (ანუ, *RAM*-ს) აიგივებენ ტერმინთან „*ოპერატიული დამხსოვებელი მოწყობილობა*“. ეს მთლად ზუსტი არ არის, რადგან არსებობს მეორე სახის თავისუფლად შელწევადი მეხსიერებაც – *მუდმივი დამხსოვებელი მოწყობილობა* (ინგლისურად – *Read Only Memory* ანუ *ROM*). *RAM*-სა და *ROM*-ს შორის მთავარი განსხვავებაა ის, რომ სამომხმარებლო ამოცანების გადაწყვეტისას *RAM*-ის შიგთავსი შეიძლება შეიცვალოს, *ROM*-ის შიგთავსი კი არა. *ROM*-ის მეხსიერების მოცულობა *RAM*-ის მეხსიერების მოცულობაზე გაცილებით მცირეა. ამის მიუხედავად, იგი კომპიუტერის მეტად მნიშვნელოვანი ნაწილია, რადგან მასში ინახება ნებისმიერ მომენტში ხელმისაწვდომი

პროგრამული უზრუნველყოფა. ამ უკანასკნელის წყალობით კომპიუტერი მაშინაც ინარჩუნებს მუშაობის უნარს, როდესაც **RAM**-ში არავითარი პროგრამა არ არის.

მაშასადამე **RAM** და **ROM** თავისუფალი შეღწევის მექანიზმების ორი სახეა, რომლებშიც შენახულ მონაცემებთან მიმართვა დამისამართების პრინციპზეა დაფუძნებული.

■ **მეხსიერების იერარქიული აგების პრინციპი.** კომპიუტერის მეხსიერებას წაყენება ორი ურთიერთსაწინააღმდეგო მოთხოვნა: მისი მოცულობა უნდა იყოს რაც შეიძლება დიდი, ხოლო მუშაობის სისწრაფე – რაც შეიძლება მალალი. ორივე მოთხოვნას ერთდროულად ვერცერთი რეალური მოწყობილობა ვერ დააკმაყოფილებს. ***მეხსიერების მოცულობის ნებისმიერი მნიშვნელოვანი გაზრდა აუცილებლად შეამცირებს მის სწრაფმოქმედებას.*** მართლაც, მეხსიერების მოცულობის გაზრდით აუცილებლად გართულება მასში საჭირო მონაცემების მოძებნა (*მაგალითად, დიდი მოცულობის მეხსიერებას დასჭირდება მრავალთანრივიან მისამართები, რაც გაზრდის კავშირის ხაზების რაოდენობას; მის შესამცირებლად იძულებული გავხდებით შევეცვალოთ დამისამართების ხერხი და, მაგალითად, მისამართი გადავცეთ ნაწილ-ნაწილ*), რაც შეანელებს მეხსიერებიდან წაკითხვის პროცესს. გარდა ამისა, სწრაფმოქმედების გაზრდა აძვირებს მეხსიერებას და მისაღები ფასის უზრუნველსაყოფისათვის იძულებული ვხდებით შევზღუდოთ ეს პარამეტრი.

მეხსიერების მოცულობასა და მის სწრაფმოქმედებას შორის არსებული წინააღმდეგობის დასაძლევად იძულებული გავხდით ერთმანეთთან იერარქიულად დაგვეკავშირებინა ორი განსხვავებული სახის მეხსიერება. თავდაპირველად ვიყენებდით მხოლოდ ორდონიანი იერარქიის მქონე მეხსიერებას, რომლის დროსაც ოპერატიული მეხსიერება დაკავშირებული იყო მაგნიტურ მავთულზე ორგანიზებული მეხსიერებასთან (ეს უკანასკნელი მაგნიტური ლენტის წინაპარია). პრაქტიკამ დაადასტურა იერარქიულად აგებული მეხსიერების ღირსება და თანამედროვე კომპიუტერებში ორზე გაცილებით მეტი დონიანი იერარქიული მეხსიერებაა გამოყენებული.

■ **პროგრამის შენახვადობის პრინციპი.** პირველ გამოთვლელ მანქანებს აპროგრამებდნენ სპეციალურ პანელებზე ზღუდარების დაყენების გზით და დაპროგრამების პროცესი რამდენიმე დღე

კრძელდებოდა. ასეთი მდგომარეობა არავის აწყობდა და **ფონ ნე-იმანის არქიტექტურაში** შემოთავაზებულ იქნა ბრძანებები ორობითი კოდეხის სახით წინასწარ ჩაგვეწერა (შეგვენახა) სპეციალურ მზიდებზე (პერფორატორებზე ან მაგნიტურ ლენტებზე) და შემდეგ შეგვეტანა კომპიუტერში. პერსონალური კომპიუტერების გამოჩენამდე მზიდებზე შენახული პროგრამის კომპიუტერში **მონაცემების მომზადების სპეციალურ მოწყობილობის** საშუალებით შეიტანებოდა. მზიდებიდან კომპიუტერში პროგრამა საკმაოდ სწრაფად შეიტანება, რითაც გამოირიცხა პროგრამის ხანგრძლივად აკრეფით გამოწვეული მანქანის მოცდენა.

პროგრამის ბრძანებებისა და მონაცემების წარმოდგენის ფორმები ვინაიდან ერთმანეთს დაემსგავსა (ორივეს მიეცა ორობითი რიცხვის ფორმა), ამიტომ შესაძლებელი გახდა ისინი ერთნაირი ხერხით შევინახოთ მეხსიერებაში. **ფონ ნეიმანმა** შემოგვთავაზა მონაცემებისა და პროგრამების შესანახად ერთი და იგივე მეხსიერება გამოგვეყენებინა, ე.ი. ისინი ერთად შეგვენახა. გამომთვლელ მანქანებს, რომელთა მონაცემები და პროგრამები ერთად ინახება, ანუ, **პრისტო-ნული არქიტექტურის** გამომთვლელი მანქანები ეწოდა.

**ჰარვარდის უნივერსიტეტში** გამომთვლელი მანქანების შექმნაზე მომუშავე ჯგუფის გადაწყვეტილებით მონაცემები და პროგრამები ერთმანეთისაგან განსხვავებულ მეხსიერებაში შეინახეს. ასეთ მანქანებს **ჰარვარდული არქიტექტურის მანქანები** ეწოდა. მთელი რიგი უპირატესობების მიუხედავად ამ არქიტექტურამ კომპიუტერებში ფეხი მკვიდრად ვერ მოიკიდა, მაგრამ იგი მასობრივად გავრცელდა მცირე მიკროპროცესორების ბაზაზე აგებულ მმართველ სისტემებში.

მეხსიერებაში მომაცემებისა და პროგრამების ერთნაირი ფორმით ჩაწერის პრინციპს **მეხსიერების ერთგვაროვნების პრინციპსაც უწოდებენ**. ამ პრინციპმა ერთი პროგრამის ბრძანებებს მეორე პროგრამის ბრძანებებით მიღების საშუალება მოგვცა. სწორედ ამის გამოა შესაძლებელი მაღალი დონის ენაზე დაწერილი პროგრამა „გადაითარგმნოს“ კონკრეტული მანქანის სამანქანო კოდეხად ანუ მოხდეს მაღალი დონის პროგრამის ტრანსლირება (გადათარგმნა) დაბალი დონის პროგრამად.

პროგრამის კოდი შეგვიძლია შევინახოთ გარე მეხსიერებაში (მაგალითად, დისკზე) და ყოველი განმეორებითი გამოთვლების საჭი-

როებისას ჩავტვირთოთ ოპერატიულ მეხსიერებაში. პროგრამების მარტივად შეცვლის გამო ელექტრონული გამომთვლელი მანქანა გადაიქცა მრავალფეროვანი ამოცანები თავისუფლად შემსრულებელ უნივერსალურ მოწყობილობად

■ **პროგრამული მართვის პრინციპი** განსაზღვრავს პროგრამის ავტომატურად შესრულების უზოგადეს მექანიზმს. გავეცნოთ მას.

**ფონ ნეიმანის მანქანაში** მართვის უმნიშვნელოვანესი ელემენტი **ბრძანებათა მისამართების მთვლეელი**. მასში ჩაიწერება შესასრულებელი პროგრამის პირველი ბრძანების ნომერი (მისამართი). პროცესორი ვალდებულია შეასრულოს აღნიშნული ბრძანება. მისი შესრულების დაწყებისთანავე მთვლეელში ჩაიწერება მოცემული პროგრამის მომდევნო ბრძანების მისამართი, რომლის შესრულებაზე პროცესორი გადავა მიმდინარე ბრძანების შესრულების შემდეგ. აღწერილი პროცესები გამეორდება პროგრამაში შემავალი ბრძანებების ამოწურვამდე, რომლის შემდეგ მიმდინარე პროგრამა ითვლება შესრულებულად; ე. ი. **პროგრამის შესრულება** ეწოდება მასში შემავალი ყველა ბრძანების შესრულებას.

მიმდინარე პროგრამის ბოლო ბრძანების შესრულების პროცესის დამთავრებამდე მთვლეელში ჩაიწერება მომდევნო პროგრამის პირველი ბრძანების მისამართი. მიმდინარე პროგრამის შესრულების შემდეგ შემდეგ პროცესორი ავტომატურად დაიწყებს მომდევნო პროგრამის შესრულებას. ასე გაგრძელდება მანამ, სანამ პროცესორი არ შეასრულებს მომხმარებლისათვის სასურველ ყველა პროგრამას, რომლის შემდეგ გამომთვლელი მანქანა გამოირთვება.

კონკრეტული პროგრამის ბრძანებებს პროცესორი ასრულებს **პროცესორის ფუნქციონირების ძირითადი ალგორითმის** შესაბამისად, რომელსაც აქვს შემდეგი სახე:

1. **ბრძანებათა მისამართების მთვლეელში** არსებული მისამართის შესაბამის მეხსიერების უჯრედიდან პროცესორი ამოიღებს შესასრულებელი ბრძანების ასლს (ანუ წაიკითხავს ამ ბრძანებას) და ჩაწერს მას და **ბრძანებების რეგისტრში**;

2. **ბრძანებების მთვლელის შიგთავსი** ისე გაიზრდება, რომ იგი გვიჩვენებდეს მორიგ შესასრულებელ ბრძანებას;

3. შესრულება **ბრძანების რეგისტრში** ჩაწერილი ბრძანება;

4. შემდეგ ზემოთ აღწერილი ბიჯები თავიდან გამეორდება.

პროგრამის ბრძანებების მიმდევრობით ავტომატურად შესრულებ-  
 ით კომპიუტერი შეასრულებს ნებისმიერ **წრფივ ალგორითმს**. პრო-  
 გრამაში **ვანშტოებისა** და **ციკლების** შესასრულებლად აუცილე-  
 ბელია დაირღვეს ბრძანებების შესრულების ბუნებრივი მიმდევრობა.  
 ამისათვის არსებობს ე. წ. **გადასვლის ბრძანებები**, რომელიც ზემოთ  
 მოყვანილი ალგორითმის მე-2 ბიჯზე **ბრძანებების მთვლეელში** შეიტ-  
 ანს ახალ შიგთავსს – **გადასვლის მისამართს**. უფრო ხშირად პრო-  
 გრამებში გამოიყენება **პირობითი გადასვლა**, ე. ი. გადასვლა მხოლო-  
 დ გარკვეული პირობის შესრულებისას მოხდება.

პროცესორის ფუნქციონირების ძირითადი ალგორითმის ასამუშა-  
 ვებლად, როგორც ზემოთ აღვნიშნეთ, **ბრძანებებათა მისამართების  
 მთვლეელში** წინასწარ უნდა იქნეს შეტანილი საწყისი მნიშვნელობა  
 – პირველად შესასრულებელი ბრძანების მისამართი. **ადრეულ გა-  
 მომთვლელ მანქანებში** ეს მისამართი ოპერატორს ხელით შეყავდა.  
**თანამედროვე კომპიუტერებისათვის** კვების მიწოდებისას მთვლეელში  
 აპარატურულად შეიტანება მუდმივ დამხსომებელ მოწყობილობაში  
 შენახული **სასტარტო პროგრამის** პირველი ბრძანების მისამართი.  
 ეს პროგრამა ჯერ ახდენს კომპიუტერის მოწყობილობების ტესტი-  
 რებას და ისინი მოყავს მუშა მდგომარეობაში, ხოლო შემდეგ ოპე-  
 რატიულ დამხსომებელ მოწყობილობაში (როგორც წესი, დისკიდან)  
 ჩატვირთავს ოპერაციული სისტემის **საწყის ჩამტვირთავს**. სწორედ  
 მას გადაეცემა შემდგომი მართვა, ხოლო მუდმივი დამხსომებელი  
 მოწყობილობიდან ჩამოტვირთული სასტარტო პროგრამა ამთავრებს  
 მუშაობას. ამ მომენტიდან დაწყებული კომპიუტერის ქცევას მასზე  
 დაყენებული პროგრამული უზრუნველყოფა განსაზღვრავს.

პროგრამის შესრულების პროცესი შეგვიძლია მნიშვნელოვნად  
 ავაჩქაროთ ალგორითმის სრულყოფის გზით. სრულყოფის იდეა ნა-  
 სესხები იქნა **კონვეიერული წარმოებიდან**, სადაც რამდენიმე მუშა  
 (ნაკეთობის საკუთარ ეგზემპლარზე) ერთდროულად ასრულებენ  
 სხვადასხვა ოპერაციას. ანალოგურად თანამედროვე მიკროპროცესო-  
 რებში ბრძანების შესრულების თითოეული ეტაპისათვის შექმნილია  
 განცალკევებული აპარატურული ბლოკი. საკუთარი ოპერაციის შეს-  
 რულების შემდეგ იგი შედეგებს გადასცემს მომდევნო ბლოკს, ხო-  
 ლო თვითონ იწყებს მორიგი ბრძანების შესრულებას.

ამ მექანიზმის არსში ყველაზე ადვილად პირველი ეტაპის – ოპერატიული დამხსომებელი მოწყობილობიდან ბრძანების ამოკრების მაგალითის განხილვით შეიძლება გავერკვეთ. **ამოკრების სპეციალიზებულ ბლოკს** მენსიერებიდან ისე ამოაქვს მიმღვერობით განთავსებული ბრძანებები, რომ არ ელოდება მათი დამუშავების დასრულებას. წაკითხული ბრძანებები განთავსდება მიკროპროცესორის შიგნით არსებულ სპეციალურ **სამუშაო მენსიერებაში**. აღნიშნულის გამო მორიგი ბრძანებების გარედან (მენსიერებიდან) შემოტანისათვის დროის დახარჯვა უკვე საჭირო არ არის, რადგან ეს ბრძანებები წინასწარ იქნა შემოტანილი პროცესორში. აღნიშნულის გამო მნიშვნელოვნად ჩქარდება პროგრამის შესრულება.

პრაქტიკაში კონვეიერული მეთოდის გამოყენება არც თუ ისე მარტივია. მაგალითად, მორიგი ბრძანების შესრულება ვერ მოხერხდება, თუ ამისათვის წინა ბრძანების შესრულების გზით მიღებული შედეგია საჭირო, ან თუ რამდენიმე ბრძანებას სჭირდება ერთდროულად მიმართოს ოპერატიულ დამხსომებელ მოწყობილობას. მიუხედავად ამისა, ეს მეთოდი ფართოდ გამოიყენება მიკროპროცესორებში. ზოგიერთ მოდელში გამოიყენება პარალელური კონვეიერები, რადგან ხშირად ერთი ბრძანების დამთავრების მომენტში უკვე მზადაა შემდგომი შედეგი.

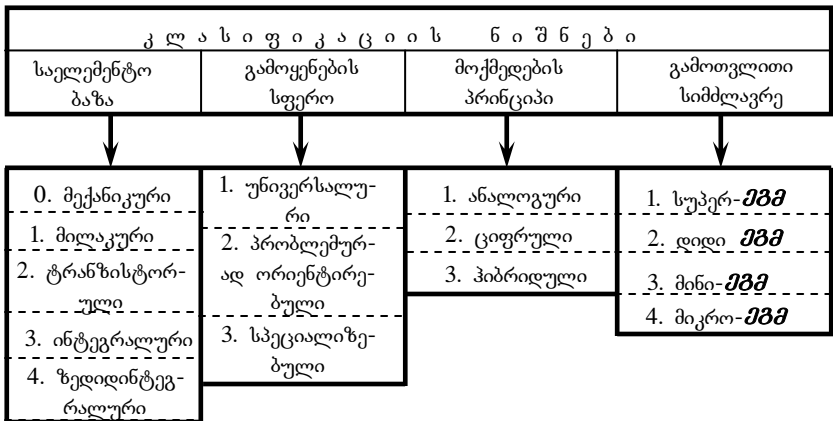
**არქიტექტურის რაობა** ფონ ნეიმანისა და მისი თანაავტორების მიერ ფორმირებული პრინციპები გამოიყენებოდა ყველა თაობის ელექტრონულ გამომთვლელ მანქანებში. აღნიშნული მანქანების თითოეულ კონკრეტულ ოჯახისათვის დამატებით ყალიბდებოდა მოწყობილობების აგების საკუთარ პრინციპები, რომლის წაყლობითაც უზრუნველყოფილი ხდებოდა მოდულების აპარატურული და პროგრამული თავსებადობა. მომხმარებლისათვის ეს ნიშნავს, რომ ნებისმიერი არსებული პროგრამის რეალიზებას შეიძლება ოჯახში შემაკალი თითოეული გამომთვლელი მანქანა. კონკრეტული კომპიუტერის აგების ზოგად პრინციპებს ლიტერატურაში **არქიტექტურა** ეწოდება. ჩვეულებრივ არქიტექტურას მიეკუთვნება: **ა)** ბრძანებათა სისტემის აგებისა და ბრძანებების კოდირების პრინციპები; **ბ)** მონაცემების ფორმატები და მათი სამანქანო წარმოდგენის თავისებურებები; **გ)** პროგრამის ბრძანებათა შესრულების ალგორითმი;

- დ) მენსიერებასთან და გარე მოწყობილობებთან შეღწევის ხერხები;
- ე) აღჭურვილობის კონფიგურაციის შეცვლის შესაძლებლობები.

საზგასასმელია ის ფაქტი, რომ არქიტექტურა აღწერს გამოთვლელი მანქანის მხოლოდ ზოგად მოწყობილობას და არა კონკრეტული კომპიუტერის დამზადების თავისებურებებს (მიკროსქემების ნაკრებს, ხისტი დისკის ტიპს. მენსიერების ტევადობას, ტაქტურ სიხშირეს).

### 1.5. გამოთვლელი მანქანების კლასიფიკაცია. ტერმინ «კომპიუტერი» დამკვიდრება

პირველი გამოთვლელი მანქანის შექმნიდან დღემდე უამრავი სხვადასხვა სახის გამოთვლელი მანქანა იქნა დამუშავებული. მათი კლასიფიკაცია შეიძლება სხვადასხვა ნიშნის მიხედვით მოვახდინოთ. ზოგიერთი ნიშნის მიხედვით გამოთვლელი მანქანების კლასიფიკაციების სქემა **1.6** ნახაზზეა მოყვანილი. გავეცნოთ მას.



**ნახ. 1.6.**გამომთვლელი მანქანების კლასიფიკაცია

საელემენტო ბაზის მიხედვით გამოთვლელი მანქანების სიმრავლე ხუთ ჯგუფად იყოფა (იხ. ნახ **1.6**), რომელთაგანაც თითოეული ჯგუფი წარმოქმნის გამოთვლელი მანქანების დამოუკიდებელ თაობას. მექანიკური ელემენტებითა და ელექტრომაგნიტური რელეებით აგებულ გამოთვლელ მანქანათა თაობას **ნულოვანი თაობა** უწ-

ოდეს, ამიტომ საელემენტო ბაზის მიხედვით გამოყოფილი ჯგუფების დანომვრა 0-დან დავიწყეთ. *პირველი, მეორე, მესამე და მეოთხე თაობის* მანქანები ეწოდება მანქანებს, რომლებიც აიგო შესაბამისად ელექტრონული მილაკებით, ტრანზისტორებით, მცირე ან საშუალო ინტეგრალური სქემებითა და ზედიდი ინტეგრალური სქემებით.

**1981** წელს *იაპონიის მთავრობამ* ხელოვნური ინტელექტის ტექნოლოგიით დამზადებული კომპიუტერებისათვის, რომელსაც *მეხუთე თაობის კომპიუტერები* უწოდა, ეროვნულ კომპანიებს **500** მილიონ დოლარს დაჰპირდა. ასეთი მანქანების აგება ჯერ-ჯერობით მსოფლიოში ცნობილმა ვერც ერთმა კომპანიან ვერ შეძლო. სამაგიეროდ ბოლო წლებში მასობრივად დაიწყო სხვა მოწყობილობებში ჩასაშენებელი მინიატურული და ზემინიატურული კომპიუტერების წარმოება, რომლებსაც ხატოვნად *უჩინარი კომპიუტერებიც* ეწოდა. შეცვლილი პარადიგმის ასეთი მანქანები ცნობილმა ამერიკელმა ინფორმატიკოსმა *ენდრიუ ტანენბაუმ მეხუთე თაობის მანქანებად* მონათლა.

გამომთვლელი მანქანების თაობებს უფრო დაწვრილებით ცალკე პარაგრაფში განვიხილავთ.

**გამოყენების სფეროს მიხედვით** გვაქვს (იხ. ნახ. 1.6):

■ **უნივერსალური *იპმ***, რომელიც განკუთვნილია ადამიანის საქმიანობის ნებისმიერი (მათემატიკის, ეკონომიკისა და ა. შ.) სფეროს ურთიერთგანსხვავებული ამოცანების გადასაწყვეტად;

■ **პრობლემურად ორიენტირებული *იპმ***, რომელიც ჩვეულებრივ განკუთვნილია ტექნოლოგიურ ობიექტებთან, აგრეთვე მცირე მოცულობის მონაცემების რეგისტრაციასა და დამუშავებასთან დაკავშირებული შედარებით ვიწრო წრის ამოცანების გადასაწყვეტად;

■ **სპეციალიზებული *იპმ***, რომელიც განკუთვნილიაწინასწარ მკაცრად განსაზღვრული ფუნქციების შესასრულებლად, რაც ამცირებს მის სირთულესა და ღირებულებას. ასეთია, მაგალითად, სპეციალური დანიშნულების კონტროლერები.

**მოქმედების პრინციპის მიხედვით** გვაქვს (იხ. ნახ. 1.6):

■ **ანალოგიური *იპმ***, რომელიც უწყვეტი მოქმედების გამომთვლელი მანქანაა; იგი აბუშავებს ანალოგიური ფორმით, ე. ი. გარკვეული ფიზიკური სიდიდის (ყველაზე უფრო ხშირად – ელექტრული



ძაბვის) მნიშვნელობების უწყვეტი მწკრივის სახით წარმოდგენილ ინფორმაციას; მათი **ღირსება** დაპროგრამების პროცესის სიმარტივე და მაღალი სწრაფმოქმედება (რომელიც, ზოგადად, დღეს ფართოდ გავრცელებული ციფრული მანქანების სწრაფმოქმედებას აღემატება). **ნაკლია** დაბალი სიზუსტე (მათი ფარდობითი ცდომილობა **2-5%**-ის ფარგლებშია) და ის გარემოება, რომ მათი საშუალებით შეზღუდული რაოდენობის ამოცანების გადაწყვეტა შესაძლებელი;

■ **ციფრული მშპ**, რომელიც ციფრული ფორმით წარმოდგენილი ინფორმაციის დამუშავებისათვის განკუთვნილი დისკრეტული მოქმედების გამომთვლელი მანქანაა. ასეთი მანქანების ღირსებაა: მაღალი სიზუსტე და ფართო წრის ამოცანების გადაწყვეტის შესაძლებლობა. ნაკლია ამოცანის დაპროგრამების პროცესის სირთულე და ანალოგურ მანქანებზე უფრო დაბალი სწრაფმოქმედება;

■ **ჰიბრიდული მშპ** - როგორც ანალოგური, ისე ციფრული ფორმით წარმოდგენილი ინფორმაციის დამამუშავებელი კომბინირებული მოქმედების გამომთვლელი მანქანაა. ანალოგური და ციფრული მანქანების ღირსებების ურთიერთშეთავსების გამო ჰიბრიდული სისტემები გამოიყენება სწრაფად მოქმედი რთული კომპლექსების მართვისათვის.

**გამოთვლითი სიმპლასრის მიხედვით** გვაქვს (იხ. ნახ. 1.6):

■ **სუაერ-მშპ**, რომელიც ერთეულოვანი ტირაჟით დამზადებულ უნიკალურ მრავალპროცესორული გამომთვლელი მანქანაა. ასეთი სისტემები მუშავდება და გამოიყენება დიდი მოცულობის მონაცემების დამუშავებასთან დაკავშირებული გარკვეული ურთულესი ამოცანების გადასაწყვეტად (კოსმოსის გამოსაკვლევა, თავდაცვის პროგრამების შესასრულებლად და ა.შ.). დღეისათვის მთელ მსოფლიოში მხოლოდ რამდენიმე ათასი ასეთი გამომთვლელი მანქანა არსებობს.

■ **დიდი მშპ**, რომლებსაც ხშირად **მეინფრეიმებსაც** (ინგლ. *Mainframe - საბაზისო*) უწოდებენ. ისინი საერთო დანიშნულების უნივერსალური სისტემებია, რომლებიც რთული სამეცნიერო-ტექნიკური და ეკონომიკური ამოცანების გადასაწყვეტად გამოიყენება. ასეთ მანქანებს სხვადასხვა საცნობარო-საინფორმაციო სისტემების ასაგებადაც გამოიყენებენ. მათი დაყენება, აწყობა და მომსახურება საინფორმაციო-გამოთვლითი ცენტრის სპეციალისტების მოვალეობაა;

■ **მინი-მზმ** ფართო წრის ამოცანების გადამწყვეტი მეოთხე თაობის გამომთვლელი მანქანებია. წინა სისტემებისაგან განსხვავებით მათ აქვს გამარტივებული სტრუქტურა და, აქედან გამომდინარე, დაბალი ღირებულება. მსგავს მანქანებს მიეკუთვნება სახლში ან წარმოებაში მუშაობისათვის განკუთვნილი პერსონალური ელექტრონული გამომთვლელი მანქანები (კომპიუტერები);

■ **მიკრო-მზმ** – სპეციალიზებული მიკროპროცესორების გამოყენებით აგებული წვრილი გამომთვლელი მანქანებია. ხშირად ისინი გამოიყენება წარმოებაში ცალკეული ტექნოლოგიური პროცესებისა და მოწყობილობების მართვისათვის. ამ კლასს შეიძლება მიეკუთვნოთ თანამედროვე დაპროგრამებადი კალკულატორები, ელექტრონული ლექსიკონები და მობილური ტელეფონები.

### ■ **ტერმინ „კომპიუტერის“ დამკვიდრების საკითხები.**

1945 წელს ტელეკომუნიკაციების, ელექტრონული კომპიუტერული სისტემების ამერიკული კომპანია **Bell Laboratories**-ის (**Bell Labs**-ის) თანამშრომელმა **ჯორჯ შტიბიცმა (1904-1995) აშშ**-ის თავდაცვის ნაციონალურ კომიტეტს წარუდგინა მოხსენება, რომელშიც ტერმინი „**კომპიუტერი**“ პირველად იქნა გამოყენებული თანამედროვე მნიშვნელობით. ზოგადად კომპიუტერი (ინგლ. **Computer** — „**გამომთვლელი**“) ოპერაციათა ზუსტად განსაზღვრული მიმდევრობის შემსრულებელი მოწყობილობა ან სისტემაა, მაგრამ **შტიბიცი** ამავე ტერმინით გამოთვლის პროცესში მონაწილე ადამიანოპერატორებსაც მოიხსენიებდა. აღნიშნული ტერმინი თანდათან ხდებოდა პოპულარული და მან წინასაუკულის **90**-იანი წლების დასაწყისიდან მთლიანად გამოდევნა ტერმინი „**ელექტრონული გამომთვლელი მანქანა**“. ჩვენთან კომპიუტერად მხოლოდ და მხოლოდ გამომთვლელ მანქანებს მიიჩნევენ, ინგლისურ ლიტერატურაში კი კომპიუტერს მათემატიკური გაანგარიშებისათვის განკუთვნილი ნებისმიერი მოწყობილობას უწოდებენ; ინგლისელებისათვის, მაგალითად, ლოგარითმული სახაზავიც კომპიუტერს წარმოადგენს.

კომპიუტერი, როგორც წესი, ნებისმიერი გამომთვლელ მანქანას აღნიშნავს, ხოლო მანქანის სახის დასაკონკრეტებლად იყენებენ ისეთ შედგენილ ტერმინებს, როგორიცაა, მაგალითად, ტერმინი „**პერსონალური კომპიუტერი**“ („**Personal computer**“).

ტერმინ „*კომპიუტერის*“ შესახებ მსოფლიოში ერთიანი დამოკიდებულება ჯერ კიდევ არ ჩამოყალიბებულა. მაგალითად, საფრანგეთში ამჟღავნებენ ტერმინს „*Ordinateur*“, რომლის პირდაპირი თარგმანია „*დამზარისხებელი*“, იტალიაში „*Computer*“—თან ერთად ზოგჯერ იყენებენ ტერმინებს „*Calcolator*“ ან „*Elaboratore*“. ფინეთში გამოთვლელ მანქანას მოუგონეს ხატოვანი სახელი „*Tietokone*“, რომლის ქართული თარგმანია „*ცოდნის მანქანა*“, ხოლო ჩინეთში — „*დიან-ნაო*“, ანუ ქართულად „*ელექტრონული ტვინი*“.

სიტყვა კომპიუტერის პირველი განმარტება ინგლისური ენის ოქსფორდისეულ ლექსიკონში 1897 წელს გამოჩნდა. ავტორი მას მქანიკურ საანგარიშო მანქანად მიიჩნევდა. 1946 წელს ლექსიკონი შეივსო დამატებებით, რომლებიც საშუალებას გვაძლევს ერთმანეთისაგან გავმიჯნოთ ციფრული, ანალოგური და ელექტრონული კომპიუტერის ცნებები.

წინა საუკუნის 70-იან წლებში ტერმინი „*კომპიუტერი*“ უპირველეს ყოვლისა *მეინფრეიმბთან (Mainframe)* ასოცირდებოდა, ხოლო პირველ მინი-*მშმ*-ებს „*დაპროგრამებად პროცესორებს*“ უწოდებდნენ. 1964 წლიდან *აშშ*-ში შემოღებული სტანდარტით აიკრძალა კომპიუტერი გვეწოდებინა ოპერატორებისათვის და იგი შეგვეძლო მხოლოდ გამოთვლელი მანქანების აღსანიშნავად გამოგვეყენებინა. აღსანიშნავია, რომ ამავე წლის 4 ნოემბერს შემოვიდა ტერმინი „*პერსონალური კომპიუტერი*“ და იგი კომპანია *Hewlett-Packard*-მა ჟურნალ *Science*-ში უწოდა მის მიერ დამუშავებულ დაპროგრამებად კალკულატორს. დღეს თვისობრივად შეიცვალა ამ ტერმინის მნიშვნელობა და იგი ეწოდა მასობრივი მოხმარებისათვის განკუთვნილ უნივერსალურ მიკროკომპიუტერს.

ის ფაქტი, რომ ტერმინი „*კომპიუტერი*“ ინგლისური ენიდან შემოვიდა, სულ არ ნიშნავს იმას, რომ იგი ინგლისური ან ამერიკული წარმოშობისაა. დაახლოებით 2000 წლის წინათ არსებობდა ლათინური სიტყვა *Computare*, რომელიც ორი ნაწილისაგან შედგებოდა — *Com* („ერთად“) და *Putare* — („დათვლა“, „მიჩნევა“, „განხილვა“ „გამოთვლა“). *VI* საუკუნეში ტერმინი *Computare* ძირითადად გამოიყენებოდა აღდგომის საზეიმო თარიღის დასადგენად საჭირო სპეციფიკური გამოთვლების აღსანიშნავად, ხოლო მისგან ნაწარმოები

სიტყვა „*კომპიუტერი*“ დღეს ზოგადად გამოთვლის ოპერაციის შემსრულებელი მანქანის აღსანიშნავად გამოიყენება (<https://masterok.livejournal.com/2284462.html>).

ტერმინი „*კომპიუტერი*“ უნდა განვასხვაოთ ტერმინისაგან „*ელექტრონული გამოძივლელი მანქანა-მზმ*“, ეს უკანასკნელი კომპიუტერის რეალიზაციის ერთ-ერთი ხერხია. *მზმ* გულისხმობს ელექტრონული კომპონენტებისაგან აგებულ კომპიუტერებს, მაგრამ ისინი სხვა კომპონენტებითაც შეიძლება აიგოს, რის გამოც არსებობს *მექანიკური, ბიოლოგიური, ოპტიკური, კვანტური* და ა. შ. გამოძივლელი მანქანები, რომლებიც შეიძლება კომპიუტერის სახელის ქვეშ გავაერთიანოთ. გარდა ამისა, ტერმინი „*კომპიუტერი*“ გულისხმობს შესასრულებელი პროგრამის შეცვლის შესაძლებლობას (გადაპროგრამებას), რისი უნარიც არ გააჩნია ყველ *მზმ*-ს.

ნებისმიერ დაპროგრამებად გამოძივლელ მანქანას ვინაიდან მოიცავს ტერმინი „*კომპიუტერი*“, ამიტომ ქვემოთ ჩვენ მხოლოდ მას გამოვიყენებთ.

## 1.6. თანამედროვე კომპიუტერების კლასიფიკაცია

დღეს არსებული თანამედროვე კომპიუტერების მთელი სიმრავლე შეიძლება შემდეგ 7 ჯგუფად დავყოთ: **1.** ჯიბის კომპიუტერები – *PDA (Personal Digital Assistant* - „პირადი ციფრული მდივანი“); **2.** პორტატული კომპიუტერები – *Notebook*; **3.** სამაგიდო კომპიუტერები – *Base PC (Personal Computer)*; **4.** მუშა სადგურები – *Workstation*; **5.** სერვერები – *Server*; **6.** სუპერკომპიუტერები – *Super Computer*; **7.** კლასტერული სისტემები – *Cluster System*.

ზემოთ ჩამოთვლილ ჯგუფებში შემავალი კომპიუტერების ღირებულებისა და მწარმოებლობის მიხედვით შეიძლება გამოვყოთ კომპიუტრთა ჯგუფების შემდეგი სამი სიმრავლე:

***I სიმრავლეში*** შემავალი კომპიუტერებისათვის უმნიშვნელოვანესი პარამეტრია ფასი. მათ უნდა შეეძლოს შეასრულოს ყველა საჭირო ფუნქცია და, იმავდროულად, ჰქონდეს მაქსიმალურად დაბალი ფასი. ამ სიმრავლის ელემენტებია *ჯიბის, პორტატული და სამაგი-*

ლო კომპიუტერები, რომლებსაც მასობრივი მოხმარების კომპიუტერებს უწოდებენ;

**II სიმრავლეში** შემაჯავლი კომპიუტერებისათვის უმთავრესი პარამეტრია მწარმოებლურობა, რომელიც რაც შეიძლება მაღალი უნდა იყოს. მათი ფასის შემცირება მიუღებელია, თუ იგი მწარმოებლურობის შემცირების მიზეზი გახდება. ეს სიმრავლე მოიცავს **სერვერებს, სუპერკომპიუტერებსა და კლასტერულ სისტემებს**. ისინი გამოიყენება ურთულესი ამოცანების გადასაწყვეტად;

**III სიმრავლეში** გაერთიანებული კომპიუტერების შექმნისას ერთდროულად უნდა გადაწყდეს მწარმოებლურობის მაქსიმალიზებისა და ფასის მინიმიზების ამოცანები; სხვა სიტყვებით რომ ვთქვათ, მათ მაღალი მწარმოებლურობა დასაშვები ფასის პირობებში უნდა ჰქონდეს (რაც ოპტიმიზაციის ამოცანას წარმოადგენს). ამ სიმრავლეში შეიძლება შევიყვნოთ **მუშა სადგურები**.

მოკლედ განვიხილოთ კომპიუტერთა ზემოთ ჩამოთვლილი ჯგუფები.

■ **ჯიბის კანონალური კომპიუტერები (PDA)**– ფართო ფუნქციური შესაძლებლობის მქონე პორტატული გამოთვლელი მოწყობილობებია. ისინი სხვა კატეგორიის კომპიუტერებთან შედარებით გაცილებით მარტივია, მაგრამ ფიჭურ ტელეფონსა და მობილურ პრინტერთან მათ სიმბიოზს სრულყოფილი მობილური ოფისის წარმოქმნა შეუძლია.

დაპროგრამებადი სწავლების დამუშავების ჩარჩოებში მიმდინარე კვლევების პროცესში 1972 წელს **აშშ**-ში გამოიგონეს **სენსორული დისკი**. ითვლება, რომ სწორედ ამ თარიღიდან უნდა დაიწყოს **PDP**-ს განვითარების ისტორია. სენსორული პანელებისა და **XX** ოს საუკუნის **80**-იან წლებში შექმნილი თხევადკრისტალური დისკების სიმბიოზის შედეგად გამოჩნდა პირველი ნოუტბუკი, რომელიც გადაიქცა **პლანშეტური კომპიუტერების** ერთ-ერთ მთავარ ნაწილად. ასეთი კომპიუტერები აღიჭურვებოდა ისეთი სპეციალური გრაფიკული ოპერაციული სისტემებით, როგორცაა **Microsoft Windows for Pen Computing, GO PenPoint, PenDOS OS/2 for Pen**.

ოპერაციულ სისტემათა დასახელებებიდანაც ნათელია, რომ ინფორმაცია შეიძლება შეგვეტანა გრაფიკული ხერხით, კერძოდ, **სტილუსად** წოდებული სპეციალური კალმით. სწორედ ასეთ მო-

წყობილობებზე მოისინჯა ხელნაწერი ტექსტის ამომცნობი სისტემა. ამ მიმართულებით დიდი ხნის განმავლობაში მუშაობდა ***Palm Computing*** (*Palm* – ხელისგული, დაუთოვება) კომპანიის დამარსებელი ***ჯეფი ჰეიკინსის*** (***Jeff Hawkins; 1957***) ჯგუფი. ამ ჯგუფმა **1989** წელს დაამზადა პლანმეტური კომპიუტერი ***GridPad***, რომელშიც ინფორმაცია კალმით შეიტანებოდა. სიძვირის, ხელნაწერის ამომცნობი სისტემის არასრულყოფილებისა და სუსტი აკუმულირების გამო ეს კომპიუტერი ვერ გავრცელდა. მიუხედავად ამისა, არ დაკარგულა კალმის მეშვეობით ინფორმაციის შეტანის იდეა და **1993** წელს ორმა კომპანიამ – ***Apple***-მ და ***Palm Computing***-მა თითქმის ერთდროულად დაამზადა ჯიბის კომპიუტერები – ***Newton Message Pad*** და ***Newton***. აქედან დაწყებული **1998** წლამდე გავრძელდა მძაფრი კონკურენცია ამ კომპანიებს შორის, რომელშიც გამარჯვება ***Palm Computing***-ს ერგო: **1997** წელს მის მიერ დამზადებულმა ჯიბის კომპიუტერმა ***Palm Professional*** საყოველთაო აღიარება პოვა და ***Apple***-მ შეწყვიტა ამ მიმართულებით მუშაობა.

***Palm Professional*** იმდენად წარმატებული, ხოლო მასში გამოყენებული ოპერაციული სისტემა ***PalmOS*** - იმდენად მარტივი და ამტანი აღმოჩნდა, რომ გიგანტმა კომპანიებმა ***Nokia***, ***Sony*** და სხვებმა, აღნიშნულ ოპერაციულ სისტემას ჯიბის საკუთარ კომპიუტერებში გამოყენების ლიცენზია მისცა.

შექმნილი სიტუაცია არ აწყობდა კორპორაცია ***Microsoft***-ს, რომელმაც დაამუშავა სპეციალურად მობილურ მოწყობილობებში გამოყენებული ოპერაციული სისტემა ***Windows CE***. მისი პირველი ორი ვერსია საკმაოდ წარუმატებელი აღმოჩნდა, მაგრამ მათთვის დამახასიათებელი ყველა ნაკლი დაძლეული იქნა **2000** წელს დამზადებულ მესამე ***Windows CE 3.0*** ვერსიაში.

ჯიბის თანამედროვე კომპიუტერებში გამოიყენება ***TFT (Thin-film transistor*** – თხელ-ფიროვანი ტრანზისტორი) ტექნოლოგიით დამზადებული დისპლეები, რომლის განრჩევადობა **320x240**-დან იწყება და რომლებსაც შეუძლია დაახლოებით **16** მილიონზე მეტი (კერძოდ, **2<sup>34</sup>**-ის ტოლი) შეფერილობის ასახვა. ჯიბის კომპიუტერები ჩვეულებრივ აღიჭურვება **64მბ**-ზე არანაკლები მოცულობის ოპერატიული მეხსიერებით, თანამედროვე მოდელებში კი ოპერატიული მეხსიერების მოცულობა **256მბ**-დან იწყება. მათი წონა არ

აჭარბებს **200** გრამს. ჯიბის ყველა თანამედროვე კომპიუტერი აღჭურვილია **SD (Secure Digital)**, **MMC (Multi Media Card)** და სხვა სტანდარტის **მეხსიერების ბარათის** მისაერთებლად საჭირო სლოტებით. მნიშვნელოვნადაა გაზრდილი უაკუმულატოროდ ჯიბის კომპიუტერის მუშაობის ხანგრძლივობა. აღვნიშნავთ, რომ **მეხსიერების ბარათი**, ანუ **ფლეშბარათი** წარმოადგენს ციფრული ინფორმაციის შესანახ კომპაქტურ ელექტრონულ დამხსომებელ მოწყობილობას. **SD** და **MMC** სტანდარტის მეხსიერების ბარათებს აქვს **4 ვივაბა-იჭამდე მოცულობის მეხსიერება**.

ჯიბის თანამედროვე კომპიუტერებს აქვს მრავალფეროვანი **მულტიმედია**: სტერეო გამოსასვლელი და პოლიფონიური დინამიკები, დიქტოფონი და **MP3**-პლეერი, ციფრული ფოტო და ვიდეოკამერა. ისინი კომუნიკაციის თვალსაზრისითაც სრულყოფილადაა აღჭურვილი: აქვს ინფრაწითელი **IrDA** პორტი, უმავთულო კავშირის **Bluetooth** პორტი და **Wi-Fi**-ადაპტერები. ინფორმაციის გადაცემის სიჩქარე უდრის – ინფრაწითელი სხივების გამოყენებისას წამში **115** კილობიტის, ხოლო **Bluetooth** პორტის გამოყენებისას – წამში **723** კილობიტის ტოლია. საკომუნიკაციო ტექნოლოგიების სწრაფმა განვითარებამ შესაძლებელი გახადა ჯიბის კომპიუტერებში გამოყენებულიყო ფიჭური კავშირგაბმულობის **GSM** სტანდარტისა და თანამგზავრული ნავიგაციის **GPS** სტანდარტის მოდულები. შევნიშნავთ, რომ **GSM** წარმოადგენს **Group Special Mobile** ჯგუფის აბრევიატურას, ხოლო – **GPM** გაიმიფრება როგორც **Global Positioning System**. ასე, რომ, ჯიბის ჩვეულებრივი კომპიუტერი შეცვალა თანამედროვე **კომუნიკატორებმა**.

„**კონიერი მობილური ტელეფონი**“, ანუ სმარტფონი (**Smart Phone**) ჯიბის შემცირებული ფუნქციური შესაძლებლობის მქონე კომპიუტერია, რომელიც წარმატებით გამოიყენება მობილურ ტელეფონად. იგი პორტატულია და შეიძლება ადვილად ჩავრთოთ ინტერნეტში.

მიუხედავად იმისა, რომ ჯიბის კომპიუტერები უკვე **20**-ზე მეტი წელია არსებობს, ბაზარზე სულ უფრო და უფრო აქტიურად გამოდის მათი ახალი ვერსიები. მათ აშუშავებს და უშვებს ისეთი ცნობილი ფირმები, როგორცაა: **Apple, Acer, HTC, Samsung** და სხვები. ამ კატეგორიის კომპიუტერების სწრაფ გავრცელებას

განაპირობებს ის გარემოება, რომ თავიანთი ფუნქციურობით ისინი სულ უფრო და უფრო უახლოვდება სამაგიდო კომპიუტერებს.

■ **კორტატული კომპიუტერები** (ნოუთბუკები). *Osborne I* სახელწოდების პირველი ასეთი კომპიუტერი **1981** წლის **3** აპრილს დაამზადა ამერიკულმა კომპანიამ *Osborne Computer Corporation*. მას ჰქონდა **5,25** დუიმი (**1** დუიმი = **2,54** სმ) ფორმატის მოქნილი მაგნიტური დისკის ორი დისკოსატარი, **5** დუიმიანი დისპლეი, იწონიდა თითქმის **12** კგ-ს და დაახლოებით **1800** დოლარი ღირდა. **1985** წელს *Nippon Shingo* კომპანიამ დაამზადა შესანიშნავი პორტატული კომპიუტერი *WS-1*, რომელიც უკვე **4** კგ-ს იწონიდა, ჰქონდა **8** მგპც სისშირის ტოლი ტაქტური სისშირე და იმარეტბოდა ოპერაციული სისტემით *BIG DOS*.

**1990** წელს უკვე გამოჩნდა ფერად ეკრანიანი მოდელი; **1991** წლიდან ეკრანად დაიწყეს თხელფიროვანი ტრანზისტორებით (*Thin-film transistor -TFT*) აგებული ფერადი აქტიური მატრიცის გამოყენება. **1994** წელს ნოუთბუკებში უკვე იყენებდნენ *Intel Pentium* ტიპის პროცესორებს.

**1995** წელს *IBM* ფირმამ დაამზადა პეპელად წოდებული გასაშლელი კლავიატურის მქონე *Think Pad 701C* მოდელის პორტატული კომპიუტერი, რომელიც დღეს ინახება თანამედროვე ხელოვნების მუზეუმში. კლავიატურის გამოჩენის პროცესი ვიზუალურად თითქმის შეუმჩნეველი იყო, რადგან ეს ხდებოდა სახურავის **150<sup>0</sup>**-ით ახდისას.

**1995-1996** წლებში სრულყოფილი ხდებოდა ნოუთბუკების გრაფიკული შესაძლებლობები (გამოჩნდა თხელკრისტალურ ეკრანზე **16** მილიონზე მეტი ფერის ასახვის უნარის მქონე *Toshiba Satellite Pro 400 CDT* მოდელი), დაიწყეს თანამედროვე (ჩამენებული *CD-ROM* ამძრავიანი *Toshiba T2150* ტიპის) დამგროვებლების გამოყენება, მცირდებოდა გაბარიტები (*Think Pad 560* ტიპის ნოუთბუკის სისქე დაახლოებით **3** სმ-ის, ხოლო წონა **2** კგ-ზე ნაკლები იყო).

**2000-2002** წლებში ნოუთბუკებისათვის გამოდიოდა *Intel*, *AMD* და *Cyrix* ფირმების მიერ დამზადებული მობილური პროცესორები; ისინი აღიჭურვებოდა *SWGA* ან *XGA* ტიპის *TFT*-დისპლეებით. კორპორაცია *Intel*-ის მიერ შემუშავებული *Mobile Power Guidelines'99* სპეციფიკაციის თანახმად ამ პერიოდში სტანდარტული



განდა **13,3**-ლიმიანი **1024x768** განრჩევალობის მქონე ეკრანი, რომელსაც შეეძლო **24** რაოდენობის შეფერილობის ასახვა. ასეთი პორტატული კომპიუტერების მასა არ აღემატებოდა **3-4** კგ-ს, ხოლო სისქე – **5** სმ-ს. ყველა მოდელი აღიჭურვებოდა **CD-ROM** (მოგვიანებით – **DVD-ROM**) ამძრავებით. ამ პერიოდის როგორც პორტატულ, ისე სამაგიდო პერსონალურ კომპიუტერებში შეიძლებოდა **Windows** ოჯახის ოპერაციული სისტემების გამოყენება.

■ **სამაგიდო კომპიუტერები** მიეკუთვნება კომპიუტერების ბაზარზე წარმოდგენილი პერსონალური კომპიუტერების ყველაზე დიდ კატეგორიას.

სამაგიდო კომპიუტერები აქტიურად გამოიყენება როგორც სხვადასხვა კორპორაციაში (**Homo PC**), ისე საოჯახო მეურნეობის სფეროშიც (**Homo PC**). ისინი შეიძლება გამოვიყენოთ როგორც სხვადასხვა თამაშით გართობისათვის, ისე რთული ამოცანების გადასაწყვეტადაც. გარდა ამისა, ასეთ კომპიუტერს შეუძლია იმუშაოს ავტომობასზე, შეადგინოს კალენდარი ან გაუძღვეს საოჯახო ბუღალტერიას, მიიღოს ელექტრონული ფოსტა, შეკვეთები გააგზავნოს მაღაზიებში. გადამწოდებთან და აღმასრულებელ მექანიზმებთან კომპიუტერის დასაკავშირებლად შეგვიძლია გამოვიყენოთ ინფრაწითელი სხივების არხი ან რადიოარხი. ასეთ კომპიუტერებს უფრო დეტალურად მეორე თავში განვიხილავთ.

■ **მუშა სადგურები** ზემოთ განხილულ კომპიუტერებზე ძვირია და გამოიყენება სპეციალურ სფეროებში, სადაც რთულ გრაფიკებთან გვიხდება მუშაობა. ასეთი კომპიუტერები გამოიყენება, მაგალითად, სამგანზომილებიანი გრაფიკის შემთხვევაში, საგამომცემლო სისტემებში და ა. შ.

მუშა სადგურებს უკავია საშუალოდ პოზიცია სამაგიდო კომპიუტერებსა და სერვერებს შორის. მათი მწარმოებლურობა დაახლოებით ქვედა დონის სერვერების მწარმოებლურობას შეესაბამება. მუშა სადგურებსა და სერვერებს შორის არსებული საზღვარი ზუსტად არ არის განსაზღვრული, ოღონდ საშუალო სადგურს ჩვეულებრივ აქვს მხოლოდ ერთი, ხოლო ქვედა დონის სერვერს – ერთი ან ორი პროცესორი.

■ **სერვერები** არაკეული ამოცანების შესასრულებლად, მაგალითად, ინფორმაციის დასამუშავებლად ან რაიმე მონაცემების მისა-

ლებად, მასთან ერთდროულად მიერთებული რამდენიმე კომპიუტერის მომსახურების უნარის მქონე სპეციალურ მაღალმწარმოებლურ კომპიუტერს ეწოდება. სერვერების მთელი სიმრავლე იყოფა მაღალი დონის (**20** ათას დოლარზე უფრო ძვირ), საშუალო დონის (**5** ათასიდან **20** ათას დოლარამდე ღირებულ) და ქვედა (საწყისი) დონის (**5** ათას დოლარამდე ღირებულ) სერვერებად.

**საწყისი დონის სერვერს** შეუძლია მოემსახუროს მცირე (**50**-მდე მომხმარებლისაგან შემდგარ) ლოკალურ ქსელს. სამაგიდო კომპიუტერისაგან განსხვავებით სერვერი **Big Tower**-ის ტიპის დიდი ზომის კორპუსს გამოიყენებს და აქვს გაფართოებისათვის საჭირო მნიშვნელოვანი რაოდენობის გასართი. სერვერების პრაქტიკულად ნებისმიერი მოდელი ხისტ მაგნიტურ დისკებზე არსებული დიდი რაოდენობის დამგროვებელის მხარდასაჭერად და მონაცემების მთლიანობის უზრუნველსაყოფად აღჭურვილია **RAID-კონტროლერებით**. საწყისი დონის სერვერზე ხშირად დაყენებულია ორი პროცესორი.

**2001** წლიდან ასეთი სერვერების დასამზადებლად გამოიყენება „ულტრამჭიდრო“ ტექნოლოგია, რომელიც ხელს უწყობს ეკონომიურ ენერგომომხმარებას და ამცირებს გაცივების სიტემათა ზომებს. შესასრულებელ ფუნქციებზე დამოკიდებულებით საწყისი დონის სერვერები იყოფა: **web**-სერვერებად, **კეშ**-სერვერებად; **SSL**-სერვერებად (**Secure Sockets Layer** – დაცული სოკეტების დონე; **სოკეტი** – ქართულად ნიშნავს გასართს.); აუდიო მონაცემებისა და ვიდეო გამოსახულებების სერვერებად; მობილური ტელეფონების ეკრანებზე ნახვისათვის გამოსადეგ ფორმატში სტანდარტული **web**-გვერდების გადამყვან სერვერებად.

**საშუალო დონის სერვერს** ჩვეულებრივ აქვს ორიდან რვაამდე სპეციალური პროცესორი, კვების ორზე არანაკლები წყარო, და ვენტილატორები; მასში გარე მეხსიერების რამდენიმე მაღალჩქაროვანი კონტროლერიცაა ინტეგრირებული.

**მაღალი დონის სერვერს** აქვს ოთხზე მეტი მაღალმწარმოებლური პროცესორი, ორზე არანაკლები რაოდენობის მუშაობის პროცესში შეცვლადი როგორც კვების წყარო, ისე ვენტილატორი და გარე მეხსიერების რამდენიმე მაღალჩქაროსნული ინტეგრირებული კონტროლერი.

სერვერების მეშვეობითაა შესაძლებელი როგორც *გლობალური*, ისე *ლოკალური* კომპიუტერული სისტემების ორგანიზება. *გლობალური კომპიუტერული* სისტემის ყველასათვის ცნობილი სახეა *ინტერნეტი*;

*ლოკალური სისტემები* ორგანიზდება ცალკეული დაწესებულებების ფარგლებში. ასეთი სისტემის მაგალითია ტექნიკური უნივერსიტეტის ფარგლებში ორგანიზებული სისტემა, რომლითაც კონტროლდება თანამშრომლების სამსახურში გამოცხადება.

სერვერების საერთო მწარმოებლურობისათვის დიდი მნიშვნელობა აქვს მეორე დონის კომპიუტერების სწრაფმოქმედებასა და მოცულობას, ამიტომ ამორჩევისას უპირატესობა უნდა მივცეთ უფრო ტევადი და სწრაფი კომპიუტერების გამოყენებით აგებულ სერვერს.

მრავალპროცესორული სისტემების გამოყენების თავისებურებების განხილვისას გასაკუთრებული ყურადღება უნდა მივაქციოთ მათ *მასშტაბურობას*, კერძოდ, შეიძლება თუ არა მათ დაეუმატოთ ახალი პროცესორები, მეხსიერების მოდული ან სხვა რესურსი. მასშტაბურობის პრაქტიკული მნიშვნელობა ვლინდება კომპიუტერული სისტემის აპარატურული და პროგრამული ნაწილების დაპროექტებისას; იგი საშუალებას გვაძლევს გავზარდოთ სისტემის მწარმოებლურობა და გამოვიყენოთ უფრო რთული ამოცანების გადასაწყვეტად.

სერვერების ერთ-ერთი უმნიშვნელოვანესი მახასიათებელია *საიმედოობა*. საიმედოობის მაღალი დონე მიიღწევა კონსტრუქციის სრულყოფითა და როგორც აპარატურული, ისე პროგრამული უზრუნველყოფის *მტყუნებისადმი ძღვრადობის* ამაღლებით. რთული მიკროპროცესორული სისტემის მტყუნებამ შეიძლება დიდი ნეგატიური შედეგები გამოიწვიოს, ამიტომ ძალიან მნიშვნელოვანია მისი საიმედოობის გაზრდა. საიმედოობა შეგვიძლია გავზარდოთ: დამატებითი (სათადარიგო) ბლოკების დაყენებით, სისტემის მყისიერი თვითრეკონფიგურაციის უზრუნველყოფით, პროგრამული უზრუნველყოფის გამართვისადმი ყურადღების ამაღლებით და ა. შ.

■ *სუპერკომპიუტერები. 1950-*იანი წლების დასაწყისიდან საშუალოდ ყოველ ორ წელში ორმაგდებოდა სამეცნიერო კვლევებისათვის განკუთვნილი დიდი კომპიუტერების სისწრაფე. სწრაფმოქმედების ამაღლებამ გაზარდა მონაცემებისა და შედეგების შესანახი

მეხსიერების ტევადობა. სუპერკომპიუტერების სიმძლავრის მაღალი დონეს განსაზღვრავდა სწრაფად განვითარებადი მიკროელექტრონიკის წარმატებები, აგრეთვე კომპიუტერის არქიტექტურის პრინციპულად ახალი კონცეფციები.

კომპიუტერული ტექნიკის არქიტექტურაში შეტანილი უმნიშვნელოვანესი სიახლეა მსგავსი ოპერაციების ერთდროულად (პარალელურად) შესრულების უნარი. **XX** საუკუნის შუა პერიოდში დამპროგრამებლები იძულებულნი იყვნენ რთული ამოცანა მთელ რიგ მიმდევრობითად შესასრულებელ ბიჯებად დაეყოთ. თანამედროვე კომპიუტერებს შეუძლია ეს პროცედურა თავად შეასრულოს, რაც ამსუბუქებს დამპროგრამებელთა საქმიანობას.

აეროდინამიკაში, სეისმოლოგიაში, ატომურ-ბირთვულ ფიზიკაში, სამხედრო კვლევებსა და მათემატიკურ მოდელირებაში შესასრულებელი ზოგიერთი გამოთვლა ზოგჯერ შეიძლება ვერც პერსონალური კომპიუტერისა და ვერც მაღალმწარმოებლური სისტემის დახმარებით ვერ შევასრულოთ. ასეთ შემთხვევაში იძულებული ვხდებით გამოვიყენოთ **სუპერკომპიუტერი**.

მოცემულ სფეროებში სუპერკომპიუტერების გამოყენების ძირითადი მიზეზია ის, რომ პროცესების მოდელირების მიზანია შევისწავლოთ გარე ზემოქმედების პირობებში ერთი წერტილიდან მეორე წერტილზე ყოველი გადასვლისას გარკვეული პარამეტრის დროზე დამოკიდებულების ხასიათი. ვინაიდან აღნიშნულ ორ წერტილს შორის უსასრულო რაოდენობის წერტილებია მოთავსებული, ამიტომ მატერიალური გარემოს სრულად აღწერისათვის უსასრულო რაოდენობის მონაცემების განხილვაა საჭირო. იძულებული ვხდებით გარემო წარმოვიდგინოთ მთელ სივრცეში განთავსებული სასრული რაოდენობის წერტილებისაგან ან კვანძებისაგან შემდგარი ბადის სახით და განვიხილოთ თუ როგორ იცვლება პარამეტრები მხოლოდ ამ ბადის კვანძებში. სიზუსტის ასამაღლებლად მაქსიმალურად უნდა გავზარდოთ აღნიშნული ბადის კვანძების რაოდენობა, რაც ასევე მაქსიმალურად გაზრდის გამოთვლების მოცულობას. ასეთი მოცულობის გამოთვლების შესრულება კი მხოლოდ ზემალაღმწარმოებლური სუპერკომპიუტერების დახმარებითაა შესაძლებელი.

მწარმოებლურობა იზომება „**მეგაფლოპებით**“, „**გიგაფლოპები**“ „**ტერაფლოპებითა**“ და „**პეტაფლოპებით**“ (ისინი შესაბამისად გვი-

ჩვენებს თუ რამდენი მილიონი, მილიარდი, ტრილიონი და კვადრალიონი ოპერაცია სრულდება წამში).

1 პეტაფლოპის გადამლახველი პირველი სუპერკომპიუტერი იყო **Roadrunner** ტაიპის კომპიუტერი, რომელიც **IBM** ფირმამ **2008** წელს დაამზადა: მისი სწრაფმოქმედება **1,042 პეტაფლოპი** იყო. დღეს ყველაზე სწრაფმოქმედი იაპონური სუპერკომპიუტერი **K computer**, რომლის სწრამმოქმედებაცაა **10,51** პეტაფლოპი.

■ **კლასტერული სისტემა (კლასტერი)** კავშირგაბმულობის მაღალჩქაროსნული არხებით გაერთიანებული კომპიუტერების ჯგუფს ეწოდება, რომელსაც მომხმარებელი ერთიან აპარატურულ რესურსად განიხილავს. კლასტერული ტექნოლოგიის ერთ-ერთმა პირველმა არქიტექტორტმა **გრეგორი ფისტერმა (Gregory F. Pfister)** კლასტერი ასე განსაზღვრა: „კლასტერი პარალელური ან განაწილებული სისტემის ნაირსახეობაა, რომელიც შედგება ერთმანეთთან დაკავშირებული რამდენიმე კომპიუტერისაგან და რომელიც ერთიან უნიფიცირებულ კომპიუტერულ რესურსად განიხილება“.

## 1.7. კომპიუტერთა თაობები და მათი დამახასიათებელი თვისებებები

კომპიუტერული ტექნოლოგიის ევოლუციის პერიოდში ასობით სხვადასხვა სახის კომპიუტერი შეიქმნა. ბევრი მათგანი უკვე დავიწყებულია, მაგრამ საერთო სიმრავლეში დღესაც ელვარებენ ისეთი კომპიუტერები, რომლებმაც წარუშლელი კვალი დატოვეს კომპიუტერული ტექნოლოგიის განვითარებაში. უდიდეს ინგლისელ მთაწროვნეს **ისაკ ნიუტონს** მიეწერება შესანიშნავი სიტყვები: „მორს იმიტომ ვიხედებოდი, რომ გიგანტების მხრებზე ვიდექიო“. კომპიუტერულ ტექნოლოგებს ასეთ არნახულ წარმატებების მიღწევაში სწორედ ამ კომპიუტერების კონსტრუქციაში განივთებული იდეებთან ზიარება ეხმარებოდა და მათი მოკლე მიმოხილვა დღესაც დიდ დახმარებას გაუწევს ახალგაზრდობას.

კომპიუტერების ამ უსასრულო სიმრავლის მიზანდასახულად განხილვის მიზნით იგი გარკვეულ თაობებად დაიყო. სხვადასხვა ავტორი მას სხვადასხვა რაოდენობის თაობად ჰყოფს. მე ვიზიარებ

კომპიუტერების მთელი სიმრავლის ექვს თაობად დაყოფის შესახებ ცნობილი სპეციალისტის *ენდრიუ ტანენბაუმის* მიერ შემოთავაზებულ იდეას, ოღონდ რამდენადმე ვაზუსტებ ნულოვანი თაობას.

**ნულოვანი თაობა  
(1623 – 1945)**

*კომპიუტერი* მონაცემების სახით წარმოდგენილი ნებისმიერი (სიმბოლური, ბგერითი, ტექსტური, გრაფიკული) სახის ინფორმაციის) ავტომატურად დამამუშავებელი მოწყობილობაა. *მონაცემები* ეწოდება ფორმალიზებული (კოდირებული) სახით წარმოდგენილ ინფორმაციას, რომლებიც გარკვეული ფორმით მეხსიერებაშია (ინფორმაციის მზიდშია) ჩაწერილი და რომლის დამამუშავება *გარკვეული ტექნიკური საშუალებებით* არის შესაძლებელი.

ინფორმაციის კოდირებისათვის შეიძლება გამოვიყენოთ თვლის ნებისმიერი სახის სისტემის მეშვეობით გამოსახული რიცხვები. ათობით რიცხვებზე ცალკეულ არითმეტიკულ ოპერაციას ასრულებს ჩვენ მიერ ზემოთ განხილული მექანიკური საანგარიშო მანქანებიც, ოღონდ მათთვის ნებისმიერი რიცხვი მხოლოდ გარკვეული სახის რაოდენობას გამოხატავს; კომპიუტერისათვის ეს რიცხვები შეიძლება გამოხატავდეს ნებისმიერი (სიმბოლური, ბგერითი, ტექსტური, გრაფიკული) სახის ინფორმაციას და მათზე იგი არითმეტიკულის გარდა, ლოგიკურ ოპერაციებსაც ასრულებს.

ზემოთ აღნიშნულიდან გამომდინარე როგორც საანგარიშო მანქანები, ისე კომპიუტერები საბოლოო ანგარიშით *რიცხვების დამამუშავებელი მოწყობილობებია*, ოღონდ კომპიუტერებში გაფართოებულია ამ რიცხვების შინაარსი, მათზე შესასრულებელი ოპერაციების მოცულობა და ოპერაციების შესრულების ტექნიკა.

ზემოთ აღნიშნულის თანახმად მექანიკური *საანგარიშო მანქანები* გარკვეული თვალსაზრისით განხილული რიცხვების პრიმიტიულად დამამუშავებელი მოწყობილობებია, რის გამოც ისინი ამერიკელმა ინფორმატიკოსმა *ენდრიუ ტანენბაუმმა* (Andrew S. Tanenbaum) *ნულოვანი თაობის კომპიუტერებად მიიჩნია* [10]. ხატოვნად რომ ვთქვათ, მექანიკურმა საანგარიშო მოწყობილობებმა შექმნა ის ნაყოფიერი ნიადაგი, რომლიდანაც კომპიუტერები აღმოცენდა.

*ნულოვანი თაობის კომპიუტერების ეპოქა* დაიწყო პირველი მექანიკური საანგარიშო მანქანის შექმნის მომენტიდან და დამთავრდა პირველი ელექტრონული მილაკური კომპიუტერების აგებით. *ენდ-*

*რიუ ტანენბაუმი* ასეთ პირველ მანქანად მიიჩნია *ბლუზ პასკალის* მიერ **1642** წელს შექმნილი საანგარიშო მანქანა, მაგრამ, როგორც ზემოთ აღვნიშნეთ, ასეთი მანქანა **1623** წელს შექმნა *შოკარდმა*, ამიტომ ნულოვანი თაობის კომპიუტერების ეპოქის დასაწყისად ეს უკანასკნელი თარიღი უნდა იყოს მიჩნეული.

ნულოვანი თაობის კომპიუტერებს, გარდა ზემოთ აღნიშნული საანგარიშო მანქანებისა, მიეკუთვნება *ჩარლზ ბებიჯის* სხვაობითი და ანალიზური მანქანები, აგრეთვე – *ჯონ ათანასოვისა* და *კლიფორდ ბერის ABC*, *ჯორჯ სტიბისის* გამომთვლელი მანქანები, *კონრად ცუზეს Z3* და *ჰოვარდ ეიკენის Mark I*.

**პირველი თაობა –  
ელექტრონული მი-  
ლაკები  
(1945 – 1955)**

მიღებული შეთანხმების თანახმად კომპიუტერების თაობები მათ ასაგებად გამოყენებული საელემენტო ბაზის ცვლილებასთან ერთად იცვლება. პირველი თაობის კომპიუტერებს აკუთვნებენ მანქანებს, რომლებშიც ძირითად ელემენტებად გამოყენებული იყო ელექტრონვაკუუმური მილაკები. აღნიშნული მილაკებით აგებულ პირველ კომპიუტერს წარმოადგენდა *ალან ტიურინგის* მონაწილეობით **1943** წელს სპეციალური სამხედრო მიზნით დამზადებული კომპიუტერი *COLOSSUS*, რომლის ასაგებად **1500** ელექტრონული მილაკი იყო გამოყენებული. იგი, როგორც ზემოთ აღვნიშნეთ, **30** წლის განმავლობაში იყო გასაიდუმლოებული და ამიტომ მას კომპიუტერული ტექნოლოგიის განვითარების პროცესში თავისი წვლილი არ შეუტანია

პირველი თაობის კომპიუტერების დამუშავებაში დიდი ღვაწლი მიუძღვის პენსილვანიის უნივერსიტეტის მეცნიერს *ჯონ მოუჩლის (John William Mauchly, 1907-1980)*. *ათანასოვის* შრომების დეტალურად გაცნობის შემდეგ მან **1942** წელს წარმოადინა გამომთვლელი მანქანის საკუთარი პროექტი. იგი მასზე მომუშავე ჯგუფს *ჯონ ეკერტისთან (1919-1995)* ერთად ხელმძღვანელობდა. მათ **1945** წლის გაზაფხულისთვის ააგეს იმ პერიოდისათვის გასაიდუმლოებული (სამხედრო დანიშნულების) პირველი მილაკური კომპიუტერი *ENIAC (Electronic Numerral Integrator And Compitr* – ელექტრონული ციფრული ინტეგრატორი და კომპიუტერი), იგი შეიცავდა სხვადასხვა ტიპის **18000** ელექტრონულ მილაკს, **1500** რელესს, **7200** კრისტალურ დიოდს, **4100** მაგნიტურ ელემენტს, ეკავა **300მ<sup>2</sup>** ფა-

რთი, იწონიდა **30** ტონას და რელეურ კომპიუტერზე **1000**-ჯერ უფრო სწრაფად მუშაობდა. მანქანაში იყო **20** რევისტრი, რომელთაგანაც თითოეულში შეგვეძლო შეგვეტანა **10**-თანრივიანი *ათობითი რიცხვი*. საიდმულოების გრიფი მას **1946** წლის **15** თებერვალს მოეხსნა.

მეორე მსოფლიო ომის დამთავრების შემდეგ გასაიდმულოების მოთხოვნა შემსუბუქდა, რის შედეგადაც *მოუჩლსა* და *ეკერტის* სკოლის გახსნის უფლება მიეცა, სადაც ისინი კოლეგა მეცნიერებს აცნობდნენ საკუთარ გამოცდილებას. ამ სკოლამ მრავალი მეცნიერი დააინტერესა ციფული კომპიუტერებით. ელექტრონული გამომთვლელი მანქანების შექმნაზე მუშაობა მკვლევრებმა მასობრივად დაიწყო. **1949** წელს კემბრიჯის უნივერსიტეტში (დიდი ბრიტანეთი) *მორის უილკის (Maurice Vincent Wilkes, 1913-2010)* ხელმძღვანელობით მომუშავე ჯგუფმა შექმნა პირველი მუშა (არა სამხედრო დანიშნულების) კომპიუტერი *EDSAC (Electronic Delay Storage Automatic Calculato)*. *ეკერტს* და *მოუჩლის* სურდათ მიეღოთ ციფრული გამოთვლითი მანქანის გამოგონებაზე პატენტი, მაგრამ სასამართლომ აღნიშნული მანქანის გამოგონების პიონერად მიიჩნია *ათანასოვი*, რომელსაც ამისათვის პატენტი არ აუღია.

*ENIAC*-ის კონსტრუქციის კრიტიკული ანალიზის საფუძველზე ებრაელი წარმოშობის უნგრელ-ამერიკელმა მათემატიკოსმა *ჯონ ფონ ნეიმანმა (1903-1957)* შემოგვთავაზა კომპიუტერების აგების პრინციპები, რომლებიც ჩვენ დეტალურად **1.4** პარაგრაფში განვიხილეთ. *ფონ ნეიმანმა ჰერმან ვოლფსტაინთან* ერთად ააგო *IAS* სახელწოდების კომპიუტერი, რომელმაც ძალიან დიდი გავლენა მოახდინა კომპიუტერული ტექნიკის შემდგომ განვითარებაზე.

დაახლოებით ამავე პერიოდში (**1951** წელს) მასაჩუსეტის ტექნოლოგიურ ინსტიტუტში (*Massachusetts Institute of Technology, MIT*) შეიქმნა რეალურ დროში მომუშავე პირველი კომპიუტერი *Whirlwind I*. ამ პროექტმა მიიყვანა ამერიკელი ინჟინერი *ჯეი ფორესტერი (Jay Wright Forrester; 1918-2016)* ჯერ მაგნიტური გულარიანი მენსიერების გამოგონებამდე, ხოლო შემდეგ პირველი სერიული მინი კომპიუტერის შექმნამდე.

მოცემულ პერიოდში *IBM* ფირმა პერფორატებისა და მათი დამახარისხებელი მექანიკური მანქანების წარმოებით იყო დაკავებული



და კომპიუტერებით არ ინტერესდებოდა. მხოლოდ **1953** წელს ააგო მან მილაკებიანი **IBM 701** კომპიუტერი, რომლის მეხსიერებაში ეტეოდა **36** ბიტიანი **2048** სიტყვა. სამი წლის შემდეგ მან შექმნა **IBM 704** სახელწოდების მორიგი კომპიუტერი, რომლის მეხსიერება იმავე რაოდენობის **4096** სიტყვას იტევდა. **1958** წელს **IBM** კომპანიამ დაიწყო მუშაობა **IBM 709** სახელწოდების უკანასკნელი მილაკიანი კომპიუტერის შექმნაზე, რომელიც **IBM 704** კომპიუტერის გაუმჯობესებული ვარიანტი იყო.

პირველთაობის კომპიუტერებს მიეკუთვნება აგრეთვე **JOHN-NIAC, ILLIAC, MANIAC, WEIZAC** და ა.შ.

პირველი თაობის კომპიუტერები ყველაზე ზუსტად შეესაბამებოდა **ფონ ნეიმანის** ძირითად პრინციპებს. მათში არსებული ერთი პროცესორი მართავდა ყველა მოწყობილობას: როგორც შიგა და გარე მეხსიერებას, ასე შეტანა-გამოტანის მოწყობილობებსაც. გარე მოწყობილობებად გამოიყენებოდა მაგნიტური ლენტისანი და მაგნიტური დოლიანი დამგროვებლები. მაგნიტურ დამგროვებლებში თავსდებოდა დაახლოებით **1,5** მეგაბაიტის, ხოლო დოლურ დამგროვებლებში - **60** კილობაიტის მოცულობის ინფორმაცია. მონაცემები და პროგრამები კომპიუტერში შეიტანებოდა პერფორაციული მზიდების – პერფობარათებისა და პერფოლენტების საშუალებით. კომპიუტერიდან ინფორმაცია გამოიტანებოდა ქალაქზე მბეჭდავი მოწყობილობის მეშვეობით.

პირველი თაობის კომპიუტერებზე მუშაობისათვის დამპროგრამებლები იყენებდნენ **სამანქანო (ორობით) კოდებს**. ამავე პერიოდში დაედო სათავე დაპროგრამების მაღალი დონის ენების განვითარებას და დაიწყო ამ ენებზე დაწერილი ბრძანებების სამანქანო ბრძანებებად გადამთარგმნი პროგრამების – **კომპილატორების** შექმნა. ტერმინი „**კომპილატორი**“ **1951** წელს პირველად შემოიღო ამერიკელმა ქალბატონმა და აშშ-ს სამხედრო ფლოტის კონტრ-ადმირალმა **გრეის ჰოპერმა (Grace Hopper, 1906-1992)**, რომელმაც დაამუშავა მათემატიკური გამოთვლების დაპროგრამების გამამარტივებელი ენიდან სამანქანო ენაზე მთარგმნელი კომპილატორი. მას ამ საბუთაოს შესრულება შესთავაზა ამერიკელმა მეცნიერმა **ჯონ ბეკუსმა (John Warner Backus, 1924-2007)**, რომელმაც მოგვიანებით (**1957** წელს)

შექმნა მაღალი დონის დაპროგრამების პირველი ენა „ფორტრანი“ (ანუ „ფორმულების ტრანსლატორი“). იგი პირველად რეალიზებულ იყო ამერიკულ კომპიუტერ **IBM 704**-ზე. შემდგომში ფართოდ გავრცელდა ამ ენის **ფორტრან-60** ვერსია. ფორტრანის ენა დღემდე რჩება ერთ-ერთ ძირითად ენად სამეცნიერო გამოთვლებისათვის.

მილაკურ კომპიუტერებს ჰქონდა დიდი ზომები, მოიხმარდა დიდი რაოდენობის (ასობითი კილოვატის ტოლ) ელექტროენერგიას, დეტალების დაზიანებების გამო ხშირად სჭირდებოდა რემონტი, იყო ძალიან ძვირი და მათ ემსახურებოდნენ ათობით და ზოგჯერ ასობით აღამიანი.

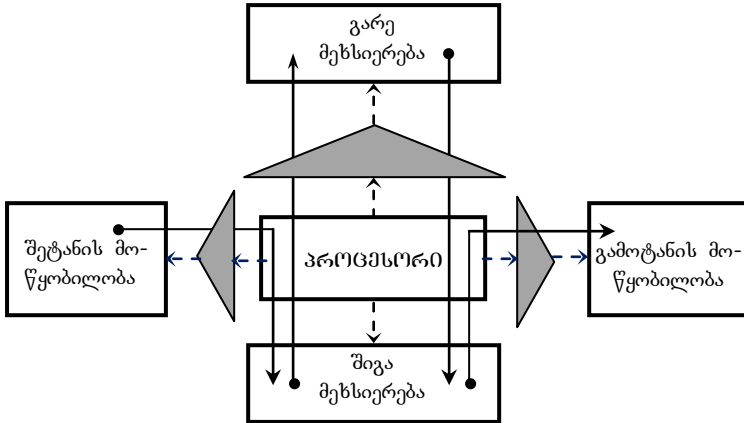
პირველი თაობის კომპიუტერები **მათზე დამპროგრამების ან ოპერატორის პირდაპირი ზემოქმედების მეშვეობით** მუშაობდა. კომპიუტერში პროგრამები და საწყისი მონაცემები პერფორმირებულიდან (პერფორმირებულიდან და პერფორმირებულიდან) შეტანის მოწყობილობის საშუალებით შეიტანებოდა. მანქანის მუშაობის პროცესში შეიძლება მართვის პულტიდან ჩართულიყო ოპერატორი. მუშაობის დასრულების შემდეგ კომპიუტერი შედეგებს დაბეჭდილი სახით გაცემდა. მუშაობის ასეთი რეჟიმის გამო მეტად დაბალი იყო კომპიუტერის მარგი ქმედების კოეფიციენტი.

**მეორე თაობა –  
ტრანზისტორები  
(1955 – 1965)**

მეორე თაობის კომპიუტერები ეწოდება ტრანზისტორებით აგებულ კომპიუტერებს. ტრანზისტორი ნახევარგამტარული ტრიოდა. ელექტრონული მილაკისაგან განსხვავებით, რომელსაც გასახურებლად გარკვეული დრო სჭირდება და ამიტომ სწრაფად ვერ გადადის მუშა მდგომარეობაში, ტრანზისტორი ჩართვისთანავე ამოქმედდება, კომპაქტურია და ელექტრულ მილაკებზე იაფია. ყველა ამ უპირატესობამ **1960**-იან წლებში განაპირობა მეორე თაობის მანქანებზე გადასვლა.

მეორე თაობის კომპიუტერების ზოგიერთ მოდელში გამეორებული იქნა პირველი თაობის კომპიუტერების არქიტექტურა და მათ შექმნეს პირველ თაობის კომპიუტერებსა თან თავსებადი კომპიუტერების ოჯახი. ამ ოჯახში შემავალ კომპიუტერებს პირველი თაობის კომპიუტერების „ტრანზისტორული გამგრძელებლები“ ეწოდა. მათი სწრაფმოქმედება პირველი თაობის კომპიუტერების სწრაფმოქმედების თანაზომადი იყო, ხოლო ოპერატიული მეხსიერების მო-

ცულობა 4-8-ჯერ აღემატებოდა პირველი თაობის კომპიუტერების ოპერატიული მეხსიერების მოცულობას. შევნიშნავთ, რომ **IBM** ფირმის მილაკური **i709** კომპიუტერის ტრანზისტორული გამგრძელებლები იყო **i7090** და **i7094**, ხოლო საბჭოური **M-20** კომპიუტერის ტრანზისტორული გამგრძელებლები - **M-220**, **M-222**, **БЭСМ-4**, **БЭСМ-6** კომპიუტერები.



**ნახ. 1.7.** ერთპროცესორიანი კომპიუტერი, რომლის გარე მოწყობილობები იმართება პერიფერიული პროცესორებით (სამკუთხედები)

გარდა ტრანზისტორული გამგრძელებლების ამ პერიოდში გამოჩნდა თვისობრივად ახალი კომპიუტერებიც, რომელთაგანაც აღსანიშნავია 1963 წელს ამერიკული კომპანია **Control Data Corporation**-ში **სეიმურ კრეის** (**Seymour Roger Cray; 1925-1996**) ხელმძღვანელობით დაშუშავებულ ორიგინალური არქიტექტურის **CDC-6600** კომპიუტერი. მასში ცენტრალურ პროცესორის გვერდით განაწილდა **შეტანა/გამოტანის არხებად** წოდებული **პერიფერიული პროცესორები**, რომლებიც 1.7 ნახაზზე მუქი ფერის სამკუთხედებითაა გამოსახული. ისინი ცენტრალური პროცესორის ნაცვლად თვითონ მართავდა **შეტანა/გამოტანის მოწყობილობებსა** და გარე მეხსიერებას. ამან მნიშვნელოვნად გაიზარდა ცენტრალური პროცესორის მარგი ქმედების კოეფიციენტი და კომპიუტერის სწრაფმოქმედებამ წამში 3 მილიონ ოპერაციას მიაღწია.

დღეს საყოველთაოდ ცნობილი **ოპერაციული სისტემები** მეორე თაობის კომპიუტერებიდან იწყებს დასაბამს. პირდაპირი დაშვების

რეჟიმის ნაცვლად იწყება *მონაცემების პაკეტური დამუშავების რეჟიმის* დანერგვა. მისი არსი ასეთია. მაგნიტურ ლენტზე ჩაიწერება რამდენიმე პროგრამა (პროგრამების პაკეტი). პროგრამებს თან დაერთვება მმართველი ბრძანებები. პროგრამისა და მმართველი ბრძანებების ერთობლიობა წარმოქმნის *დავალებას*; ეს უკანასკნელი განსაზღვრავს პროგრამაზე განასახორციელებელი მოქმედებების თანამიმდევრობას, კერძოდ, თუ როდის უნდა მოხდეს მოხდეს პროგრამის შეტანა, ტრანსლატორის გამოძახება და ამუშავება, მონაცემების შეტანა, შესასრულებლად პროგრამის ამოქმედება და ა. შ.

კომპიუტერში დავალება შეიტანებოდა მაგნიტური ლენტიდან. ერთი დავალების შესრულებისთანავე კომპიუტერის მოუცდენელად იწყებოდა მეორე დავალების შესრულება. მუშაობა მთავრდებოდა შედეგების დაბეჭდვით. დავალებების ნაკადს მართავდა სისტემური პროგრამა, რომელსაც დავალებების პაკეტურად დამამუშავებელი *მონიტორი (დისპეტჩერი)* ერქვა.

ამ თაობის პერიოდში წარმოიშვა და გაფართოვდა მცირე კომპიუტერების წარმოება. დიდ კომპიუტერებთან შედარებით ისინი უფრო მარტივ და იაფ მანქანებს წარმოადგენდა. ასეთი მცირე კომპიუტერებიდან უნდა გამოვეყთ *DEC* კომპანიის მიერ შექმნილი *PDP-1* და *PDP-8* ტიპის კომპიუტერები.

*PDP-1* კომპიუტერი *1961* წელს გამოჩნდა. მის მენსიერებაში *18* ბიტიანი *4096* ეტეოდა და მას შეეძლო წამში *200000* ოპერაციის შესრულება. ეს ორჯერ ჩამოუვარდებოდა *7090* ტიპის კომპიუტერის სწრაფმოქმედებას, მაგრამ მისი ფასი *120000* დოლარი იყო მაშინ, როდესაც *7090* კომპიუტერი მილიონობით დოლარი ღირდა. *DEC* ფირმამ შეძლო ათობით ასეთი კომპიუტერი გაეყიდა და ამით საფუძველი ჩაეყარა კომპიუტერულ მრეწველობას.

რამდენიმე წლის შემდეგ *DEC* კომპანიამ დაამუშავა *PDP-1*-ზე იაფი *12*-თანრივიანი *PDP-8* კომპიუტერი: მისი ფასი იყო *16000* დოლარი. მასში შეტანილი მთავარი სიახლე იყო ერთადერთი სალტის (*Omnibus*-ის) გამოყენება. ასეთი არქიტექტურა დღემდე გამოიყენება ყველა მიკრო- და პერსონალურ კომპიუტერებში. *DEC* კომპანიამ გაყიდა *PDP-8* მოდელის *50000* კომპიუტერი და პირველი ადგილი დაიკავა მინიკომპიუტერების ბაზარზე. საბჭოურ მინი-კომპიუტერებს წარმოადგენდა მანქანები *Промин*, *Мир* და *Наури*.

ამ პერიოდში მაღალი შედეგები იქნა მიღწეული **დაპროგრამების ავტომატიზაციის** სფეროში. მუშაობა ძირითადად მიმართული იყო მაღალი ღონის **დაპროგრამების ენის დამუშავებისა** და მათთვის **ტრანსლატორების შექმნისაკენ**; დიდი ყურადღება ექცეოდა აგრეთვე **პროგრამების გამართვის სისტემების სრულყოფასა** და **სტანდარტ-პროგრამათა ბიბლიოთეკების** დამუშავებას. ევროპულ ქვეყნებში ვითარდებოდა და ვრცელდებოდა დაპროგრამების ენა **ალგოლი**, **აშშ**-ში – **ფორტრანი** და **პობოლი**. **1964** წელს **აშშ**-ში დამწყები დამპროგრამებლებისათვის შექმნეს დაპროგრამების ენა **ბეისიკი**, რომელიც შემდეგ წარმატებით იქნა გამოყენებული პერსონალურ კომპიუტერებში. გამოჩნდა გარკვეული კლასის ამოცანებზე ორიენტირებული არაპროცედურული ენები: **ლისპი** – ფუნქციური დაპროგრამებისათვის, **სნობოლი** – სტრიქონების დასამუშავებლად, **სიმულა-1** – მოდელირებისათვის და ა.შ.

**მესამე თაობა – ინტეგრალური სქემები და მრავალამოცანურობა (1965 – 1980)**

საბჭოთა აკადემიკოსი ნ. ნ. მოისეევი აღნიშნავდა, რომ «ელექტრონული გამოთვლელი ტექნიკის ნამდვილი ისტორია და საზოგადოებაზე ამ ტექნიკის რევოლუციური ზემოქმედება იწყება **60**-იანი წლებიდან, როდესაც გამოჩნდა ე. წ. **მესამე თაობის ელექტრონული გამოთვლელი მანქანები**». ამ თაობის კომპიუტერების საელემენტო ბაზად გამოყენებული იქნა ინტეგრალური სქემები, რომელთა სამრეწველო წარმოება **1962** წელს დაიწყო ამერიკულმა კომპანიამ **Intel** («**INTEGRAL ELECTRONICS**»).

**1964** წელს **IBM** კომპანიამ განაცხადა ინტეგრალური სქემების მეშვეობით ახალი თაობის კომპიუტერების **6** მოდელის აგების შესახებ. ამ მანქანების სერიაში მიიღო სახელწოდება **IBM/360**. ისინი იყო ერთნაირი არქიტექტურის პროგრამულად თავსებადი მანქანები, რომლებიც ერთამანეთისაგან განსხვავდებოდა მწარმოებლობით, კომპლექტაციითა და ფასით. სხვადასვა მოდელების სწრაფმოქმედება მოთავსებული იყო წამში რამდენიმე ათეული ათასი ოპერაციის შესრულებიდან - რამდენიმე მილიონამდე ოპერაციის შესრულებამდე.

**1970** წელს დაიწყო **IBM/370** სერიის კომპიუტერების დამზადება. ისინი სრულიად განსხვავებულ, მაგრამ **IBM/360**-თან პროგ-

რამულად თავსებად კომპიუტერებს წარმოადგენდა. მათ ასაგებად გამოყენებული იყო დიდი ინტეგრალური სქემები, ე. წ. **დის**-ები.

გარე მოწყობილობების მართვისათვის **IBM/360/370** სერიის კომპიუტერების კონსტრუქციაში არსებობდა ერთი ცენტრალური და რამდენიმე პერიფერიული პროცესორი. ამ უკანასკნელებს ეწოდებოდა **შეტანა/გამოტანის არხები**. პერიფერიული პროცესორების შემოტანამ შესაძლებელი გახადა რეალიზებულიყო **მუშაობის მულტი-პროცესორული რეჟიმი**. ამ რეჟიმის დროს მანამ, სანამ ერთი პროგრამა მონაცემების შეტანა/გამოტანის ოპერაციის შესრულებითაა დაკავებული, მეორე პროგრამა ცენტრალური პროცესორის მეშვეობით კონკრეტულ გამოთვლებს ასრულებს.

მესამე თეობის კომპიუტერთა **ოპერაციულ სისტემებს უნდა შეესრულებინა შეძლევი 3 ამოცანა:**

**1.** კომპიუტერის რესურსები ისე უნდა გაენაწილებინა ერთდროულად შესასრულებელ რამდენიმე პროგრამას შორის, რომ ერთდროულად მოქმევე პროგრამებს ერთმანეთისათვის „არ შეემალა ხელი“ და მაქსიმალური ყოფილიყო პროცესორის მარგი ქმედების კოეფიციენტი (არ უნდა მომცდარიყო პროცესორი). პროგრამებს შორის, უპირველეს ყოვლისა, ნაწილდებოდა პროცესორის მუშაობის დრო და ოპერატიული მეხსიერება. ოპერაციული სისტემა რამდენიმე პროგრამის მიერ საერთო გარე მოწყობილობების - გარე მეხსიერების, შეტანა/გამოტანის მოწყობილობების - გამოყენების წესის დაცვასაც იღებდა თავის თავზე.

**2.** რადგან მესამე თაობის კომპიუტერებში გარე მეხსიერებად მიმდევრობითად შეღწევადი მაგნიტური ლენტისანი დამგროვებლის ნაცვლად გამოყენებული იყო თავისუფლად შეღწევადი მაგნიტური დისკისანი დამგროვებელი, ამიტომ ოპერაციული სისტემის საშუალებით უნდა მომხდარიყო მაგნიტურ დისკზე არსებულ ფაილებთან მუშაობის პროცესის მართვა.

**3.** მესამე ამოცანა იყო კომპიუტერთან მომხმარებლის (კომპიუტერის ოპერატორის, დამპროგრამებლის) ურთიერთმოქმედების უზრუნველყოფა. ეს ურთიერთმოქმედება ხორციელდებოდა ოპერაციული სისტემის სპეციალური საკომანდო ენის გამოყენებით. ურთიერთშემოქმედების ტექნიკურ საშუალებად (ინტერფეისად) თავდაპირველად გამოიყენებოდა კონსოლები - ელექტრული საბეჭდი მანქანები, ხო-

ლო მოგვიანებით – ელექტრონულ-სხივური დისპლეთი დამზადებულ მონიტორები.

ყველაზე განვითარებული შესაძლებლობის იყო ოპერაციული სისტემა *OS/360*. მის საფუძველზე იქნა რეალიზებული კომპიუტერის *მუშაობის მრავალსამომხმარებლო რეჟიმი*. ერთ მანქანაზე რამდენიმე მომხმარებელს ერთდროულად შეეძლოთ შეესრულებინათ *დისკულიანი კლავიშური მოწყობილობის სახის ტერმინალით* შეტანილი საკუთარი პროგრამა. თვისობრივად ახალ დონეზე დაბრუნდა პირველი თაობის კომპიუტერების დრო, როდესაც მომხმარებელი (დამპროგრამებელი) პირდაპირ ურთიერთზემოქმედებდა კომპიუტერთან; სიასლე იყო ის, რომ უკვე ბევრ მომხმარებელს შეეძლო ერთდროულად (პარალელურად) ემუშავა. *სატერმინალო დარბაზები* სამანქანო დარბაზისაგან განცალკევებულად ეწყობოდა.

კომპიუტერზე მუშაობის ძირითად საშუალებად რჩებოდა დაპროგრამების მაღალი დონის ენები. ფართოდ გავრცელდა ნებისმიერი ტიპის მონაცემებთან სამუშაოდ განკუთვნილი დაპროგრამების ენა *PL-1 (Programming Language-1)*. წარმოების, ეკონომიკისა და მეცნიერების სხვადასხვა სფეროსათვის მუშავდებოდა *გამოყენებითი პროგრამების მრავალი პაკეტი*. კომპიუტერთან კავშირის ტერმინალური სისტემით ისინი შეეძლოთ გამოეყენებინათ დაპროგრამების უცოდინარ მომხმარებლებს.

*IBM/360/370* სერიის კომპიუტერების არქიტექტურა ფაქტობრივად მესამე თაობის დიდი მანქანებისთვის საერთაშორისო სტანდარტად გადაიქცა. სხვა ქვეყნებში დაიწყო *IBM/360/370*-თან შეთავსებადი მანქანების დამზადება. მათ ამზადებდა დიდი ბრიტანეთის, *ბზრ*-ის, იაპონიის, ჰოლანდიის. საბჭოთა კავშირში ასეთ კომპიუტერს წარმოადგენდა ერთიანი *EC* სერიის კომპიუტერები. ყველა მათგანში გამეორებული იყო *IBM/360/370* სერიის კონსტრუქციული და პროგრამული გადაწყვეტები.

*1960*-იანი წლების ბოლოს *DEC* კომპანიამ დაიწყო *PDP* სერიის კომპიუტერების დამზადება. ამ პატარა მანქანებს *შეეძლო რეალური დროის რეჟიმში მუშაობა* და ამის წყალობით სხვადასხვა *ტექნიკური მოწყობილობის მართვა*. ამიტომ მათ *მართველ მანქანებსაც* უწოდებდნენ. ასეთ მანქანებს შორის სახელი ყველაზე მეტად *PDP-II* ტიპის კომპიუტერმა გაითქვა. ასეთი სერიის მანქანების არქიტე-

ქტურას, როგორც ზემოთ აღვნიშნეთ, *საერთო სალტის არქიტექტურა* ეწოდება. საერთო სალტე წარმოადგენს საინფორმაციო მაგისტრალს, რომელთანაცაა მიერთებული პროცესორი, მეხსიერება და გარე მოწყობილობათა კონტროლერები. მოგვიანებით ასეთი არქიტექტურა გამოიყენეს მიკროკომპიუტერებსა და პერსონალურ კომპიუტერებში.

მინიკომპიუტერების პროგრამულ უზუნველყოფას საფუძვლად დაედო *რეალური დროის ოპერაციული სისტემები*. ისინი არა მარტო მულტიპროგრამულ რეჟიმში მანქანის რესურსების განაწილების პროცესს მართავდა, არამედ გარე ზემოქმედებებზე კომპიუტერის რეაქციის დროსაც განსაზღვრავდა. მინიკომპიუტერების გამოყენება კოსმოსური ხომალდების ბორტებზეც დაიწყო. მათ ბაზაზე დაამუშავდა *ტექნოლოგიური პროცესების მართვის ავტომატიზებული სისტემები*.

თანდათანობით წაიშალა მცირე და დიდი კომპიუტერების საექსპლუატაციო თვისებებს შორის არსებული საზღვარი. **1970-1980** წლებში *IBM*-ში გამოდიოდა *VAX* სერიის **32**-თანრიგის, მილიონიანი სწრაფმოქმედებისა და გიგაბაიტის ტოლი მეხსიერების მქონე მცირე კომპიუტერები. იმ პერიოდში შეეძლოთ ასეთი მანქანები სუპერ მანქანების კლასისთვის მიეკუთვნებინათ.

*მმრთხე თაოზა – ზმ-დიდი ინტეგრალური სემევი (1980 - ?)*

კომპიუტერების მეოთხე თაობაზე გადასვლა ორ მოვლენასთანაა დაკავშირებული; პირველი მოვლენა იყო სუპერკომპიუტერის გამოჩენა, ხოლო მე-

ორე მოვლენა - *მიკროპროცესორების გამოგონება და კომპიუტერებში მათი გამოყენება*. ორივე ეს მოვლენა **1970**-იანი წლების დასაწყისში მოხდა და შესაძლებელი გახდა კომპიუტერების ასაგებად *ზედიდი ინტეგრალური სქემების* გამოყენების წყალობით (ზედიდი ინტეგრალური სქემა ეწოდება სქემას, რომლის კრისტალში **10000**-ზე მეტი ელემენტია თავმოყრილი. *მცირე, საშუალო* და *დიდი ინტეგრალური სქემის* კრისტალში შესაბამისად **100**-მდე, **1000**-მდე და **10000** ელემენტია მოთავსებული).

პირველი **i4004** ტიპის მიკროპროცესორი ფირმა *Intel*-მა **1971** დაამუშავა. ეს **4**-თანრიგიანი მიკროპროცესორი წაშში **60000** ოპერაციას ასრულებდა. **1973** წელს ამავე ფირმამ დაამზადა **8**-თანრიგ-



რიგიანი *i4008* ტიპის მიკროპროცესორი, რომლის საფუძველზე მოგვიანებით დამუშავებული იქნა პირველი მიკროკომპიუტერი. შემდგომში შეიქმნა მიკროკომპიუტერების მთელი ოჯახები, რომელთა თანამედროვე წარმომადგენელი – *პერსონალური კომპიუტერი* დღეს არნახული პოპულარობით სარგებლობს. მიკროკომპიუტერებსა და პერსონალური კომპიუტერებს მეტ-ნაკლებად ვრცლად მეორე თავში განვიხილავთ, ახლა კი მთელ ყურადღებას *სუპერკომპიუტერებზე* გადავიტანთ.

კომპიუტერების გამომგონებლებისათვის თავიდანვე იყო ნათელი, რომ ერთადერთი „გამომთვლელი“ პროცესორიანი კონსტრუქციის მქონე მანქანის სწრაფმოქმედება უსასრულო ვერ იქნებოდა და აუცილებლად იარსებობდა ზღვარი, რომლის გადაღახვა შეუძლებელი იქნებოდა. ეს შეუძლებლობა ორი მიზეზით არის განპირობებული; *ჯერ ერთი*, კომპიუტერის კვანძებს შორის სიგნალის გავრცელების სიჩქარეს აქვს გარკვეული მაქსიმალური სასრული სიდიდე *და, მეორეც*, აღნიშნული მაქსიმალური სიდიდის მიღწევაც შეუძლებელია პროცესორის მუშაობის სიჩქარის პროპორციულად სიბოროს გამოყოფის ზრდის გამო: პროცესორის ტემპერატურამ შეიძლება ისეთ სიდიდეს მიაღწიოს, რომელიც მას უბრალოდ დაწვავს. წამში ათობითი მილიონი ოპრაციის შესრულების რეჟიმში მომუშავე პროცესორის ტემპერატურა უკვე უახლოვდება ასეთ ზღვარს.

გამოსავლის ძებნის პროცესში კონსტრუქტორებმა დაასკვნეს, რომ საჭირო იყო, *უარი ეთქვათ კომპიუტერის ნეიმანისეულ ერთ-პროცესორულ არქიტექტურაზე* და შეიმუშავეს ამ არქიტექტურის მოდერნიზების ორი ფუნდამენტური პრინციპი; ესენია *კონვეიერული დამუშავებისა* და *ვექტორული დამუშავების პრინციპები*. ამ პრინციპებისა და სხვა ინოვაციური მეთოდების გამოყენებამ იმდენად აამაღლა კომპიუტერების ტექნიკური მაჩვენებლები, რომ მათ *სუპერკომპიუტერები* უწოდეს. მიღებული განმარტების მიხედვით *სუპერკომპიუტერი* ეწოდება ისეთ სპეციალიზებულ გამოთვლელ მანქანას, რომელიც საკუთარი ტექნიკური პარამეტრებისა და გამოთვლების სიჩქარის მიხედვით მნიშვნელოვნად აღემატება მსოფლიოში არსებული კომპიუტერების უმრავლესობას.

ჯერ გავცნოთ *კონვეიერულ მეთოდს*. ყოველ ჩვენთაგანს გვაქვს წარმოდგენა, მაგალითად ავტომობილების ასაწყობ კონვეიერის შე-

სახე. აწყობა შედგება მიმდევრობითი ოპერაციებისაგან, რომელთაგანაც თითოეულს თითო მუშა ან რობოტი ასრულებს. კონვეიერზე გადანაცვლების კვალობაზე ნაკეთობა თანდათან იღებს საბოლოო სახეს. ახლა წარმოვიდგინოთ, რომ კონვეიერი არ არსებობს და ავტომობილის ასაწყობად საჭირო ყველა სამუშაოს ერთადერთი უნივერსალური მანქანა ასრულებს. ცხადია, რომ კონვეიერული რეჟიმის დროს დღეში უფრო მეტი ავტომობილი აეწყობა, ვიდრე ერთადერთი უნივერსალური მანქანის გამოყენების დროს.

კომპიუტერში თითოეული ოპერაციის (ბრძანების) შესრულება ცალკეულ ეტაპებად იყოფა. *პროცესორულ კონვეიერს* წარმოქმნის პროცესორების მწკრივი, რომელთაგანაც თითოეული მათგანი მხოლოდ ერთ ოპერაციას ასრულებს და შედეგს აწვდის კონვეიერში ჩაბმულ მეორე პროცესორს. ამის შემდეგ იგი იწყებს შემდეგი ოპერაციის საკუთარი ეტაპის შესრულებას.

მაგალითად, ვთქვათ, რომ თითოეული ბრძანების შესრულება იყოფა **4** ეტაპად და კონვეიერში არსებული თითოეული პროცესორი დროის ერთ ერთეულში მხოლოდ საკუთარ სამუშაოს ასრულებს. **4** ერთეული დროის შემდეგ კონვეიერიდან „ჩამოვა“ **1**-ლი ბრძანება, **5** ერთეული დროის შემდეგ – მე-**2** ბრძანება და ა.შ. **N** რაოდენობის ბრძანების შესასრულებლად საჭირო იქნება დროის **N + 3** ერთეული. ახლა დავუშვათ, რომ ბრძანებებს ასრულებდეს ერთადერთი პროცესორი და მას თითო ოპერაციის შესასრულებლად დროის **4** ერთეული სჭირდებოდა. ასეთი მანქანა **N** რაოდენობის ბრძანებებს დროის **4N** ერთეულში შეასრულებდა.

*კონვეიერული პრინციპი* არა მარტო სუპერკომპიუტერებში, არამედ **i80486** პროცესორიდან დაწყებული ჩვეულებრივ პერსონალურ კომპიუტერებშიც გამოიყენება.

*ვექტორული დამუშავებისას* გამოიყენება პარალელურად მომუშავე პროცესორული ელემენტები, რომლებიც სხვადასხვა მონაცემზე ერთსა და იმავე ოპერაციებს ასრულებს. მსგავსი გამოთვლები ხშირად სრულდება რიცხვთა მასივების – ვექტორების დამუშავებისას. მაგალითად, **100** ელემენტებიანი ერთგანზომილებიანი  $\{a_1, a_2, \dots, a_{100}\}$  და  $\{b_1, b_2, \dots, b_{100}\}$  მასივის შეკრებისას მიიღება იმავე ზომის მასივი, რომლის ელემენტები გამოითვლება ფორმულით:  $c_i = a_i + b_i$ ,  $i = 1, \dots, 100$ . ამ ამოცანას თუ გადაწყვეტს ერთპროცესორუ-

ლი კომპიუტერი, მაშინ შეკრებები მიმდევრობით შესრულდება და გამოთვლებზე დაიხარჯება  $100\tau$  დრო, სადაც  $\tau$  არის შეკრების ერთ ოპერაციაზე დახარჯული დრო. კომპიუტერში თუ პროცესორების რაოდენობა  $100$ -ზე არანაკლები იქნება, მაშინ ისინი პარალელურად იმუშავენ და მთელ სამუშაოს  $\tau$  დროის განმავლობაში დაასრულებს, ე.ი. სამუშაო  $100$ -ჯერ უფრო სწრაფად შესრულდება.

ვექტორებზე ოპერაციების შესრულებისას მუშაობის კონვეიერული პრინციპის გამოყენებაზე დაფუძნებული *ვექტორულ-კონვეიერული კომპიუტერი*.

*მატრიცული კომპიუტერი* მხოლოდ ვექტორული პრინციპით მუშაობს. ამ დროს პროცესორული ელემენტები სინქრონულად ასრულებს ოპერაციების ერთსა და იმავე მიმდევრობას.

*მრავალპროცესორული კომპიუტერებში* თითოეული პროცესორული ელემენტი სხვებისაგან დამოუკიდებლად საკუთარი პროგრამით მუშაობს. სხვა ელემენტებს იგი უკავშირდება მონაცემების გადაცემის გზით. ეს გადაცემა კომპიუტერის მეხსიერების გავლით ხორციელდება. პარალელურად მომუშავე პროცესორებს შორის მონაცემების სწრაფად გადაცემის პრობლემა მრავალპროცესორული კომპიუტერის საკვანძო პრობლემაა.

სუპერკომპიუტერების შექმნა  $1970$ -იანი წლებიდან დაიწყო. დიდი წარმატებებით სარგებლობდა სუპერკომპიუტერების მწარმოებელი ფირმა *Cray Research*-ის დამაარსებლის ამერიკელი კონსტრუქტორის *სეიმურ კრეის (1925-1996)* მიერ დამუშავებული მანქანები.  $1976$  წელს მან გამოუშვა ნამდვილ რიცხვებზე წამში  $130$  მილიონი ოპერაციის შემსრულებელი *CRAY-1* სუპერკომპიუტერი, რომელშიც გამოთვლების *ვექტორულ-კონვეიერული ტექნოლოგია* იყო გამოყენებული.

წამში  $200$  მილიონი ოპერაციის შემსრულებელი ერთ-ერთი პირველი *მატრიცული სუპერკომპიუტერია 1972* წელს ილინოისის უნივერსიტეტში შექმნილი მანქანა *LLLIAC-IV*.

$1980$ -იან წლებში *კრეი* დაკავებული იყო *მრავალპროცესორული სუპერკომპიუტერების* კონსტრუირებით. *CRAY-2*, *CRAY-s* და სხვა მოდელის სწრაფმოქმედებამ წამში  $10^9$ - $10^{10}$  ოპერაციას მიაღწია ამ პერიოდში *იაპონიაში* მაღალი ტემპით ვითარდებოდა სუპერკომპიუტ-

ერების წარმოება. იაპონური სუპერკომპიუტერი წამში **20** მილიარდ ოპერაციისა ასრულებდა.

**1980-1990** წლებში დამზადებული მრავალპროცესორული კომპიუტერების შემადგენლობაში შედიოდა რამდენიმე ერთეული ან ათეული პროცესორი. ერთეულიდან თანამედროვე მრავალპროცესორული სუპერკომპიუტერი შეცავს **ათასობით** პარალელურად მომუშავე პროცესორს. მათი სწრაფმოქმედება წამში **10<sup>12</sup>** ოპერაციის თანაზომადია.

მაღალპროცესორული სუპერკომპიუტერი ძალიან ძვირია. პარალელური გამოთვლების უზრუნველყოფის უფრო იაფი ხერხია **განაწილებული გამოთვლების ტექნოლოგია**.

**განაწილებული გამოთვლები** ქსელის სახით გაერთიანებული კომპიუტერების სიმრავლის მეშვეობით პარალელური გამოთვლების ორგანიზების ხერხია. ასეთ გამოთვლით სისტემებს **მულტიკომპიუტერული სისტემებიც** ეწოდება.

განაწილებული გამოთვლები ხშირად რეალიზდება **კომპიუტერული კლასტერების** მეშვეობით. **კლასტერი** ეწოდება ლოკალურ ქსელში არსებულ რამდენიმე კომპიუტერს, რომლებიც გაერთიანებულია გამოთვლითი პროცესის მარეალიზებელი პროგრამული უზრუნველყოფის მეშვეობით.

დღეისათვის პარალელური გამოთვლების ორგანიზების ყველაზე იაფი საშუალება კლასტერული სისტემებია, რადგან მათ შესაქმნელად შეიძლება გამოვიყენოთ უკვე ჩვენს მფლობელობაში არსებული კომპიუტერები; ოღონდ უნდა აღვნიშნოთ, რომ მრავალპროცესორულ სისტემებთან შედარებით ცალკეული კომპიუტერების ურთიერთდაკავშირებით წარმოქმნილი სისტემების არც შესაძლებლობები და არც სწრაფმოქმედება მაღალი არ არის.

ბუნებრივად იბადება კითხვა, რა საჭიროა ზესწრაფი კომპიუტერები. მიხედავად იმისა, რომ კომპიუტერების მწარმოებლურობა ყოველ **4-5** წელში პრაქტიკულად ორმაგდება, მაინც არსებობს ისეთი კლასის ამოცანები, რომლებსთვისაც არაა საკმარისი ცალკეული კომპიუტერის მწარმოებლურობა. ასეთი ამოცანებია:

■ ბუნებასა და ტექნიკაში არსებული მრავალი პროცესის მათემატიკური მოდელების ასაგებად საჭირო გამოთვლები. გიგანტური გამოთვლითი რესურსებია საჭირო ამინდის საიმედო და ხანგრძლი-

ვი ვადით პროგნოზირებისათვის, აეროკოსმოსური, მათ შორის, სამხედრო ამოცანების გადასაწყვეტად, მრავალი საინჟინრო ამოცანების შესასრულებლად და ა. შ.

■ რამდენიმე ტერაბაიტის მოცულობას მიღწეულ ( $1$  ტერაბაიტი =  $1024$  გიგაბაიტი –  $10^{12}$  ბაიტი) გიგანტურ მონაცემთა ბაზებში ინფორმაციის მოძებნა;

■ ინტელექტის მოდელირება. თანამედროვე კომპიუტერების ოპერატიული მეხსიერების მოცულობა, მიუხედავად ერთი შეხედვით ფანტასტიკურად სიდიდისა, ადამიანის მეხსიერების მოცულობის მხოლოდ მცირე ნაწილია.

**მეხუთე თაობა – მცირე სიმპლავრისა და უზინარი კომპიუტერები**

**1981** წელს იაპონიის მთავრობამ გადაწყვიტა **500** მილიონი დოლარი გამოეყო იმ ნაციონალური კომპანიებისათვის, რომლებიც შექმნიდა ხელოვნური ინტელექტზე დაფუძნებულ

კომპიუტერებს; ასეთ კომპიუტერებს მან მეხუთე თაობის კომპიუტერები უწოდა და მათ უნდა შეევიწროვებინა მეოთხე თაობის კომპიუტერები. იაპონელი კონსტრუქტორებისათვის რომ დაესწროთ, ამერიკელმა და ევროპელმა მწარმოებლებმა ამ საქმეში დასახმარებლად სუბსიდიების თხოვნა დაუწყეს თავის მთავრობებს. ხმაურიანი დასაწყისის მიუხედავად გარკვეული დროის შემდეგ ნათელი გახდა მეხუთე თაობის კომპიუტერების დამუშავების **იაპონური იდეის** არარეალობა და იგი წყნარად მოიხსნა დღის წერიგიდან. თვისობრივად ახალ დონეზე გამეორდა **ბებიჯის** დროინდელი სიტუაცია, როდესაც იდეამ ომდენად გაასწრო დროს, რომ რეალობაში არ აღმოჩნდა მისი განხორციელებისათვის აუცილებელი ტექნოლოგიური ბაზა.

სრულიად მოულოდნელი ფორმით მოხდა ისეთი კომპიუტერების რეალიზება, რომლებსაც **მეხუთე თაობის კომპიუტერები** შეიძლება ეწოდოს. კერძოდ, კომპიუტერის ზომებმა საოცარი სისწრაფით დაიწყო შემცირება. **1989** წელს ფირმა **Gard**-მა გამოუშვა **GridPad**-ის სახელწოდების პლანშეტური კომპიუტერი. მას ჰქონდა მომცრო ეკრანი, რომელზეც მომხმარებელს სპეციალრი კალმით წერა შეეძლო. **GridPad**-ის მსგავსმა კომპიუტერებმა ნათლად დაგვანახა, რომ სრულებით არ არის საჭირო კომპიუტერი იდეგს მაგიდაზე ან

სასერვეროში – მომხმარებელს შეუძლია იგი ატაროს, ხოლო სენსორულმა ეკრანმა და ხელნაწერი ტექსტის აღქმამ ასეთი კომპიუტერი უფრო მოსახერხებელი გახადა.

**1993** წელს გამოჩენილმა *Apple Newton* მოდელმა თვალნათლივ დაამტკიცა, რომ კომპიუტერი სრულიად შესაძლებელია მოთავსდეს კასეტური პლეერის ზომის კორპუსში. *GridPad*-ის მსგავსად *Newton*-ში ინფორმაცია ხელნაწერის სახით შეიტანებოდა; ამან თავდაპირველად რამდენადმე შეაფერხა მისი გავრცელება, მაგრამ შემდგომში *პერსონალური ელექტრონული მდივნებად (Personal Digital Assistant - PDA)*, ანუ უბრალოდ, *ჯიბის კომპიუტერებად* წოდებული მსგავსი მანქანების სამომხმარებელი ინტერფეისი სრულყოფილი იქნა და მან ფართო პოპულარობა მოიპოვა. ევოლუციის მორიგ ეტაპად დღეს მოგვევლინა *სმარტფონები*.

*PDA*-ში კალმის მეშვეობით ინფორმაციის შეტანის ინტერფეისი სრულყო მასობრივი მოხმარებისათვის განკუთვნილი ჯიბის საქმიანი კომპიუტერების დამამუშავებელი კომპანია *Palm*-ის დამაარსებელმა *ჯეფ ჰოკინსმა*. იგი განათლებით ინჟინერი იყო, მაგრამ ძალიან ინტერესდებოდა ნეირობიოლოგიით (მეცნიერება ადამიანის ტვინზე). მან დაადგინა, რომ ხელნაწერი შეტანის საიმედოობის ასამაღლებლად საჭირო იყო მომხმარებლისათვის გვესწავლებინა მის მიერ დამუშავებული ე. წ. „*Graffiti*“ სახელწოდების ტექნოლოგია. აღნიშნული ტექნოლოგია ადვილი შესასწავლი იყო, ხოლო მისი საშუალებით შესრულებულ ხელნაწერს კომპიუტერი ადვილად აღიქვამდა. *Palm*-ის მიერ დამუშავებულმა ჯიბის პირველმა *Palm Pilot* კომპიუტერმა მომხმარებელთა დიდი მოწონება დაიმსახურა, ხოლო კომპიუტერულ ტექნოლოგიაში ერთ-ერთ საუკეთესო მიღწევად მიჩნეულმა „*Graffiti*“ ტექნოლოგიამ დამაჯერებლად დაამტკიცა, რომ პრაქტიკაში წარმატებით შესაძლებელია გამოყენებული იყოს ნებისმიერი გენიალური გამოგონება

*PDA*-ს მფლობელები აღმერთებდნენ საკუთარ მოწყობილობას და გულმოდგინედ იყენებდნენ მას საკუთარი განრიგისა და კონტაქტების მართვისათვის. **1990**-იან წლებში ძალიან გავრცელდა ფიჭური ტელეფონები. *IBM* ფირმამ ფიჭური ტელეფონი ჩააშენა *PDA*-ში და ამ გზით შექმნილ მოწყობილობას „*სმარტფონი*“ ეწოდა. *Si-mon*-ად წოდებულ პირველ სმარტფონში ინფორმაციის შესატანად

გამოიყენებოდა სენსორული ეკრანი. მისი წყალობით მომხმარებელს შეეძლო გამოეყენებინა როგორც **PDA**-ის, ასევე ფიჭური ტელეფონის, ელექტრული ფოსტის, ყველა შესაძლებლობა, თავი გაერთო თამაშებით. მცირე ზომებისა და დაბალი ღირებულების გამო სმარტფონები ძალიან სწრაფად გავრცელდა. დიდი პოპულარობა მოიპოვა **Apple iPhone** და **Googl Android** პლატფორმებმა.

**ცხრ. 1.1.** კომპიუტერების თაობების ზოგადი დახასიათება

თაობები	max სწრაფ-მოქმედება (ოპერ./წმ)	ოპერატიული მეხსიერების max ტევადობა (Kბაიტი)	არქიტექტურა
0-ოვანი თაობა	კალკულატორებისა და ENIAC-ამდელი კომპიუტერების თაობა		
I თაობა	10-20 ათასი	100	ნეიმანისეული ერთპროცესორული
II თაობა	100 ათასი – 3 მილიონი	1000	ნეიმანისეული ერთპროცესორული. პერიფერიული პროცესორების გამოჩენა
III თაობა	100 მილიონი	10000	ცენტრალური პროცესორი + შეტანა/გამოტანის არხები; სალტური არქიტექტურა
IV თაობა	$10^9$ - $10^{12}$	$10^7$	კონვეიერულ-ვექტორული, მატრიცული, მრავალპროცესორული. მულტიკომპიუტერული სისტემები
V თაობა	<b>პარადიგმას შეცვლა:</b> მცირე ზომისა დაუჩინარი კომპიუტერების თაობა		

ყოველივე ზემოთ აღნიშნულის მიუხედავად ჯიბის კომპიუტერებიც ვერ გადაიქცა ნამდვილ რევოლუციური ნაკეთობებად. გაცილებით მნიშვნელოვანი აღმოჩნდა საყოფაცხოვრებო ტექნიკაში, საათებში, საბანკო ბარათებში და უამრავ სხვა მოწყობილობაში ჩასაშენებელი ე.წ. „**უჩინარი**“ კომპიუტერები. ამ ტიპის პროცესორებს აქვს ფართო ფუნქციური შესაძლებლობები და, საკმაო ზომიერი ფასის პირობებში, გამოეყენების ვარიანტების არანაკლებ ფართო სპექტრი. ჩამოთვლილი კომპიუტერები **1970**-იანი წლებიდან გამოჩნდა და მნიშვნელოვნად გააფართოვა ათასობით საყოფაცხოვრებო და სხვა მოწყობილობების შესაძლებლობები; მიუხედავად ამისა, ჯერ კიდევ გადაუწყვეტია ერთ თაობად მათი გაერთიანების საკითხი. მსოფლიო

მრეწველობის განვითარებაში უჩინარი კომპიუტერების გავლენა ძალიან დიდია და წლების გასვლით თანდათან იზრდება. ამგვარი კომპიუტერების თავისებურებაა ის, რომ მათი აპარატურული და პროგრამული უზრუნველყოფა ხშირად *თანდაშუშავების* მეთოდით პროექტდება.

ცხრილ 1.1-ში მოცემულია კომპიუტერების ჩვენს მიერ შემოთავაზებული თაობების ზოგადი დახასიათება

**რეზიუმეს მაგიერ** საბოლოოდ შეიძლება დავასკვნათ, რომ **პირველი თაობის** მიეკუთვნება მილაკური, *მეორე თაობის* – ტრანზისტორული, ხოლო *მესამე თაობის* – ინტეგრალური კომპიუტერები; *მეოთხე თაობის* ფაქტობრივად შეიძლება პერსონალური კომპიუტერები მივაკუთნოთ. რაც შეეხება *მეხუთე თაობას*, ისინი უფრო ასოცირდება არა კონკრეტულ არქიტექტურასთან, არამედ *პარადიგმის შეცვლასთან*. მომავლის კომპიუტერები ყველა მოაზრებად მოწყობილობაში ჩაშენდება და ამით მართლაც უჩინარ კომპიუტერებად გადაიქცევა. ისინი მტკიცედ დამკვიდრდება ყოველდღიურ ცხოვრებაში – გაგვიღებს კარს, ჩაგვირთავს ნათურას, გაგვინაწილებს ფულს, მოგვემსახურება მაღაზიაში და ათასობით სხვა ფუნქციას შეგვისრულებს. ამერიკელი ინფორმატიკოს *მარკ ვაიზერის (Mark D. Weiser, 1952 – 1999)* მიერ დამუშავებულმა ამ მოდელმა მიიღო *საყოველთაო კომპიუტერიზაციის* სახელწოდება, რომელსაც დღეს *ყველგან შეღწეად კომპიუტერიზაციასაც* უწოდებენ. იგი მსოფლიოს სამრეწველო რევოლუციაზე უფრო რადიკალურად შეცვლას უქადის.



## II ტავი პერსონალური კომპიუტერი

### 2.1. პერსონალური კომპიუტერების ისტორია და არქიტექტურა

*პირველად იყო ... მიკროპროცესორი!*

**პერსონალური კომპიუტერი** (Person-Computer, PC) ეწოდება საყოფაცხოვრებო ხელსაწყოს მახასიათებლებისა და

უნივერსალური ფუნქციური შესაძლებლობების მქონე სამაგიდო **მიკროკომპიუტერს**. ეს უკანასკნელი, თავის მხრივ, ისეთი კომპიუტერია, რომელშიც ცენტრალურ პროცესორად მიკროპროცესორია გამოყენებული.

ზემოთ აღნიშნულიდან გამომდინარე, მიკროპროცესორის მეშვეობით იქნა აგებული მიკროკომპიუტერი, ხოლო ეს უკანასკნელი მამინ გარდაიქმნა პერსონალურ კომპიუტერად, როდესაც მოხერხდა საყოფაცხოვრებო ხელსაწყოთვის დამახასიათებელი მახასიათებლებით მისი აღჭურვა!

კომპიუტერების მესამე თაობამდე პროცესორი კომპიუტერთან ერთად აიგებოდა: მრეწველობა პროცესორებს დასრულებული პროდუქციის არ აწარმოებდა. მესამე თაობის კომპიუტერების ასაგებად ინტეგრალური სქემების გამოყენების წყალობით მოხერხდა პროცესორის ფუნქციების შემსრულებელი მობილური მიკროსქემის შექმნა, რომელსაც მცირე გაბარიტული ზომებს გამო **მიკროპროცესორი ეწოდება**. აქვე აღვნიშნავთ, რომ მიკროპროცესორის თვისებებს შორის მთავარია არა მარტო მისი მცირე ზომები, არამედ ის, რომ იგი მობილურობის უნარის მქონე პროცესორია. ვინაიდან დღეს ნებისმიერი პროცესორი მიკროპროცესორის სახით მზადდება, ამიტომ ხშირად ტერმინებს „პროცესორსა“ და „მიკროპროცესორს“ სინონიმებადაც ვიყენებთ.

მიკროპროცესორის გამოყენებით აგებულმა კომპიუტერმა *მიკროკომპიუტერის* სახელწოდება მიიღო, რომელიც **1980**-იანი წლების ბოლომდე დომინირებდა; **1990**-იანი წლების დასაწყისიდან ზემოთ აღნიშნული ცვლილების გამო იგი შეცვალა სახელმა „*პერსონალური კომპიუტერი*“. დღეს ტექნიკურ ლიტერატურაში ორივე ტერმინი გამოიყენება.

პერსონალური კომპიუტერის გენეზისის სათავეში, როგორც ზემოთ აღვნიშნეთ, დგას მიკროპროცესორი, რაც იმის მტკიცების საშუალებას გვაძლევს, რომ პირველად იყო ... მიკროპროცესორი! აღნიშნულის გამო მოკლედ შევხებით მისი წარმოშობის ისტორიას.

წინა საუკუნის **60**-იან წლების ბოლოს კალკულატორების მწარმოებელმა იაპონურმა კომპანია *Busicom*-მა გადაწყვიტა შეექმნა გაუმჯობესებული ტექნიკური მაჩვენებლების მქონე დაპროგრამებადი კალკულატორი. მის ასაგებად საჭირო იყო ორიგინალური კონსტრუქციის მქონე სხვადასხვა სახის **12** მიკროსქემა. ამ პერიოდში მიკროსქემებს ამზადებდა სპეციალიზებული ფირმები, რომელთა შორის მაშინ (და ახლაც!) ერთ-ერთ მოწინავედ ითვლებოდა ფირმა *Intel* და სწორედ მას შეუკვეთეს აღნიშნული მიკროსქემების დამზადება. *Intel*-ის ხალმძღვანელობამ ამ შეკვეთის შესრულება დაავალა თავიანთ თანამშრომლებს *ტედ ჰოფს (Ted Hoff)*, *ფედერიკო ფაჯჯინის (Federico Faggin)* და *სტენ მეიზორს (Sten Mazor)*. ეს სამეული დასმული ამოცანის გადაწყვეტას ინოვაციურად მიუდგენ და წინადადება წამოაყენეს ერთმანეთისაგან განსხვავებული **12** მიკროსქემის ნაცვლად დაემუშაებინათ კაჟბადის ერთ კრისტალზე განთავსებული უნივერსალური მიკროსქემა, რომელიც თორმეტივე მიკროსქემის ფუნქციას შესრულებდა და ცენტრალური პროცესის მსგავსად იმუშავებდა. მცირე ზომებისა და ცენტრალურ პროცესორთან მსგავსების გამო მას მიკროპროცესორი უწოდეს. მათი წინადადება ფირმამ მოიწონა, რის შემდეგ *ჰოფმა* დაამუშავა მიკროპროცესორის არქიტექტურა, *მეიზორმა* – ბრძანებათა სისტემა, ხოლო *ფაჯჯინმა* თვით მიკროსქემა დააპროექტა. მათი რეალიზების გზით **1971** წლის **15** ნოემბერს ფირმა *Intel*-მა საკვეთად წარმოადგინა *i4004*-ად წოდებული **2300** ტრანზისტორისაგან აგებული პირველი **4**-თანრივიანი მიკროპროცესორი, რომლის ტაქტური სიხშირე **108** კილოჰერცი იყო. *i4004*-ის შექმნა არა მარტო გამოთვლითი

ტექნიკის სპეციალისტებისათვის, არამედ რიგითი ადამიანებისათვის ხელმისაწვდომი *პერსონალური კომპიუტერის* შექმნისაკენ გადადგმული პირველი ნაბიჯი იყო. **1972** წლის აპრილში ფირმა *Intel*-მა დაამზადა **0,5** მეგაჰერციანი ტაქტური სიხშირის **8**-თანრივიანი მიკროპროცესორი **i4008**. მის ბაზაზე **1993** წელს *Intel*-მა შეიქმნა *Intellec-8* სახელწოდების პირველი *მიკროკომპიუტერი*.

პირველ *მიკროკომპიუტერებზე* მუშაობისათვის აუცილებელი იყო დაპროგრამირების ცოდნა, რის გამოც მათ მასობრივ მომხმარებელთა შორის წარმატება ვერ მოიპოვა. ამ კომპიუტერების ფასიც არ იყო მანკდამანკ დაბალი – იგი **2500** დოლარამდე ღირდა. გასა-იაფებლად მწარმოებლებს აზრად მოუვიდათ მოეხინჯათ „კომპიუტერ-კონსტრუქტორის“ შექმნის იდეა: მომხმარებლებს შესთავაზეს დეტალების ნაკრები და მათი აწყობის ინსტრუქცია, რომლის დახმარებითაც მას მირჩილვის წესით უნდა შეერთებინა დეტალები და მოეხდინა აწყობილი კვანძების ტესტირება. ამ კომპიუტერებს არ ჰქონდა არც კლავიატურა, არც დისკლეი და არც ხანგრძლივი დროით შემნახველი მეხსიერება. პროგრამები მომხმარებელს წინა პანელზე არსებული ტუმბოვების გადართვების გზით თავად უნდა შეეტანა, ხოლო შედეგები შუქდიოდური ინდიკატორებიდან წაეკითხა. კომპიუტერ-კონსტრუქტორებს წარმოადგენდა *Micral, SCE-LBI-8H, Mark 8* ტიპის მიკროკომპიუტერები. მათი ღირებულება იყო დაახლოებით **500-600** დოლარი, მაგრამ ასეთი დაბალი ფასის მიუხედავად, მომხმარებლები გულგრილნი დარჩნენ მათ მიმართ. პირველი მიკროკომპიუტერი, რომელმაც მიიქცია მომხმარებელთა ყურადღება იყო ამერიკული კომპანია *MITS (Micro Instrumentation and Telemetry Systems)*-ის მიერ **1975** წელს დამზადებული კომპიუტერი *Altair 8800*, რომელშიც გამოყენებული იყო ფირმა *intel*-ის მიერ დამზადებული **i8080** ტიპის მიკროპროცესორი. მისმა შემქმნელმა *ჰენრი ედვარდ რობერტსმა (Henry Edward Roberts; 1941-2010)* კომპიუტერში გამოიყენა სისტემური დაფა, რომლის ბუდეებში თავსდებოდა პროცესორი, მეხსიერება და სხვა საჭირო მოწყობილობები. მასში რეალიზებული იქნა *კომპიუტერის ღია არქიტექტურის პრინციპი*. ღია არქიტექტურის დროს საქვეყნოდ ცხადდება კომპიუტერის სპეციფიკაცია და კომპიუტერული აქსესუარების დამამზადებელ სხვადასხვა ფირმას კომპიუტერის სრულყოფის პროცეს-

ში მონწილეობის შესაძლებლობას ეძლევა. ეს აუმჯობესებს სისტემის როგორც ტექნიკურ, ისე ეკონომიკურ მაჩვენებლებს.

**პერსონალური კომპიუტერის „დაბადება“**

ტერმინ «პერსონალური კონსტრუქტორის» გამოჩნა დაკავშირებულია ამერიკელი სპეციალისტების *სტივენ პოლ (სტივ) ჯობსის (Steven Poul “Steve” Jobs; 1955-2011)*, *სტივენ გერი (სტივ) ვოზნიაკის (Stephen Gary „Steve“ Wozniak; 1950)* სახელებსა და მათ მიერ დაარსებულ კომპანია *Apple*-თან (ინგ. *Apple* - ვაშლი), სადაც **1976** წელს შეიქმნა ადრეული პერსონალური კომპიუტერი *Apple-1*, ხოლო **1977** წელს - *Apple-2*.

არსებული მიკროკომპიუტერისაგან პერსონალური კომპიუტერი ძირითადად იმით განსხვავდება, რომ მასზე მუშაობა შეუძლია არა მარტო გამოთვლითი ტექნიკის სპეციალისტებსა და დამპროგრამებლებს, არამედ ნებისმიერი სხვა პროფესიის მომხმარებელს, მათ შორის მოსწავლეებსა და სტუდენტებსაც. პერსონალური კომპიუტერის მოწყობილობების შემადგენლობაში გაითვალისწინება გრაფიკული დისპლეი, მოსახერხებელი კლავიატურა და პრინტერები. **1978-1979** წლებში აღნიშნულ კომპიუტერებში გამოჩნდა გამოყენებითი პროგრამული უზრუნველყოფა – *ტექსტური რედაქტორი WordStart* და *ცხრილური პროცესორი VisiCalc*.

პერსონალური კომპიუტერების პოპულარობამ რამდენადმე შეამცირა მოთხოვნილება დიდ და მინი-კომპიუტერებზე. ამან სერიოზულად შეაშფოთა კომპიუტერების ძირითადი მწარმოებელი კომპანია *IBM* და მან **1979** წელს გადაწყვიტა ჩართულიყო პერსონალური კომპიუტერების წარმოების პროცესში. ფულადი სახსრების დასაზოგად კომპანიის ხელმძღვანელობამ პერსონალური კომპიუტერების წარმოებაზე პასუხისმგებელ ქვეგანყოფილებებს ნება დართო კონსტრუირებისათვის სხვა ფირმების მიერ დამუშავებული ბლოკებიც გამოეყენებინა, რამაც კომპიუტერულ სამყაროში საბოლოოდ დაამკვიდრა «*Altair 8800*» კომპიუტერის შემქმნელების მიერ შემოტანილი *პერსონალური კომპიუტერის ღია არქიტექტურის პრინციპი*: პერსონალური კომპიუტერები გარდაიქმნა და დღემდე რჩება ღია არქიტექტურის მოწყობილობებად.

საკუთარი პირველი პერსონალური კომპიუტერი *IBM* ფირმამ **1981** წლის **12** აგვისტოს დაამზადა და მას *IBM PC* უწოდა. ამ

კომპიუტერის პირველ **5150** მოდელში მან გამოიყენა **Intel**-ის მიერ დამუშავებული **4,77** მგჰც ტაქტური სიხშირიანი **i8088** მიკროპროცესორი. ოპერატიული მეხსიერების მოცულობა **64** კილობაიტის ტოლი იყო. კომპიუტერზე არ იყო გარე (ხანგრძლივი დროის) მეხსიერება. ამ პროცესორისათვის ფირმა **Microsoft**-მა დაამუშავა ახალი ოპერაციული სისტემა **MS-DOS**. კომპიუტერზე შეიძლებოდა **10** ან **20** მეგაბაიტის მოცულობის ხისტი დისკის დაყენება. მან სწრაფად მოიპოვა დიდი პოპულარობა და დამზადების პირველსავე წელს **136000** ასეთი კომპიუტერი გაიყიდა. ეს ფანტასტიკურ წარმატებად ჩაითვალა, რის გამოც ზემოთ აღნიშნული თარიღი **პერსონალური კომპიუტერის „დაბადების“** თარიღად იქნა მიჩნეული. უფრო მეტიც, **1981** წლის მსოფლიოს ნომერ პირველ ადამიანად პირველად (და უკანასკნელადაც!) არა რომელიმე გამოჩენილი მოღვაწე, არამედ პერსონალური კომპიუტერი იქნა მიჩნეული. დაჯილდოების ცერემონიაზე ჟურნალ „**Time**“-ის გამომცემელმა **ჯონ მეერმა** ეს ასე ახსნა: „**1981** წლის ჯილდოს კანდიდატად შეიძლება რამდენიმე ადამიანი წარგვედგინა, მაგრამ ვერც ერთი მათგანის ღვაწლი ვერ შეედრებოდა იმას, რაც შეძლო საყოველთაოდ ხელმისაწვდომ კომპიუტერმა“.

**1984** წელს იქნა დამზადდა **PC AT** (**A**dvanced **T**echnoljgy – *სრულყოფილი ტექნოლოგია*) სახელწოდების მომდევნო მოდელი. ამ მოდელში გამოყენებული იყო **20მგჰც**-მდე სიხშირიანი **i80286** პროცესორი. ამით დამთავრდა კომპანიის ლიდერობა. **i80386** პროცესორის ბაზაზე აგებული „**IBM**-შეთავსებადი“ პირველი კომპიუტერი დაამზადა **Compaq Computers** კომპანიამ. იგი იყო პირველი **32**-თანრივიანი პროცესორი, რომელმაც დაულო სათავე **IA-32** (**32-bit Intel Architecture**) არქიტექტურის მქონე პროცესორების ოჯახს.

**1984** წელს ფირმამ **Apple** დაამზადა **Macintosh** (*ინგ. Macintosh წყალკაუმტარი, რეზინიანი მატერია*) სახელწოდების პერსონალური კომპიუტერი, რომელშიც საკომანდო სტრიქონიანი ინტერფეისის ნაცვლად პირველად იქნა გამოყენებული ამერიკელი გამომგონებლის **დუგლასენგლბარტის** (**Douglas Carl Engelbart, 1925-2013**) მიერ გამოგონებული გრაფიკული სამომხმარებლო ინტერფეისი. ამ ინტერფეისთან მუშაობისათვის ასევე პირველად დაიწყო **ენგლბარტის** მი-

ერ 1968 წლის 9 მაისს გამოგონებული *მაუსის* ფართო გამოყენება (იგი მან 1970 წელს დააპატენტა). აღანიშნავია, რომ *ენგელბერტის* მიერაა გამოგონებული *ჰიპერტექსტი* (*ჰიპერტექსტი* აღნიშნავს ჯვარდის მიმართვიან ტექსტურ ვკერდებს) და *ტექსტური რედაქტორი* (*ტექსტური რედაქტორი* ეწოდება ზოგადად ტექსტური მონაცემებისა და კერძოდ ტექსტური ფაილების შესაქმნელად განკუთვნილ დამოუკიდებელ პროგრამულ კომპლექსს ან ამ კომპლექსის კომპონენტს).

თავდაპირველად *Macintosh*-ის სახელწოდების კომპიუტერებში გამოყენებული იყო მიკროპროცესორები *Motorola*, რომლებიც შემდეგ შეიცვალა *IBM*-ის მიერ დაბეჭადული უფრო მძლავრი *PowerPC* მიკროპროცესორით. 2006-წლიდან *Apple* ფირმა გადავიდა *Intel*-ის მიერ გამოშვებულ მიკროპროცესორებზე. აღსანიშნავია ის ფაქტი, რომ კონკურენტებისაგან განსხვავებით, საკუთარი კომპიუტერებისათვის იგი თვითონვე ქმნის *Mac OS* სახელწოდების ოპერაციულ სისტემას.

#### მიკროპროცესორების ევოლუცია

პირველი მიკროპროცესორის დამზადებიდან დღემდე ფირმა *Intel*-ი რჩება მსოფლიოში მიკროპროცესორების დამამზადებელ ერთ-ერთ წამყვან კომპანიად. ცოტა მოგვიანებით მისი კონკურენტი გახდა ფირმა *AMD* (*Advanced Micro Devices* – „მოწინავე მიკრომოწყობილობები“). დღეს სწორედ ისინია პერსონალური კომპიუტერებისათვის გათვალისწინებული მიკროპროცესორების დამამზადებელი ძირითადი ფირმები. გარკვეული პერიოდის განმავლობაში მათ კონკურენციას უწევდა ისეთი ფირმები, როგორიცაა *Apple Computer*, *IBM* და *Motorola*.

მიკროპროცესორული ტექნიკის განვითარებაში უმნიშვნელოვანეს ეტაპის დასაწყისი გახდა 1989 წელი, როდესაც *Intel*-მა გამოუშვა მიკროპროცესორი *i80486*. ამ პროცესორის მიერ შემოთავაზებულ პირველ სიახლეს წარმოადგენდა *ცენტრალურ პროცესორში* მცურავი რეგისტრების არითმეტიკული ოპერაციების შემსრულებელი *მათემატიკური თანაპროცესორის* ინტეგრირება. გარდა ამისა, ამ მიკროპროცესორში რეალიზებული იყო კონვეიერი, რომელიც მანამდე მხოლოდ სუპერკომპიუტერებში იყო გამოყენებული.

1993 წელს *Intel*-მა დაამზადა მიკროპროცესორი *Pentium*, რომელშიც პირველად განხორციელდა გადასვლების წინასწარმტყვე-

ლება. **გადასვლების წინასწარმეტყველების** არსია ის, რომ პირობითი გადასვლის (მაგალითად, განშტოების) ბრძანების შესრულებისას პირობის გაანალიზებამდე **მიკროპროცესორის სპეციალური ბლოკი** ან საზღვრავს გადასვლის ყველაზე მოსალოდნელ მიმართულებას. პროცესორი მეხსიერებიდან იწყებს პროგრამის ნაწინასწარმეტყველები განშტოების შესაბამისი ბრძანების ამორჩევასა და შესრულებას. რადგან შესაძლებელია, რომ განშტოების მიმართულება არასწორად ყოფილიყო ამორჩეული, ამიტომ მიღებული შედეგები ჩაიწერება არა მეხსიერებაში, არამედ **რეგისტრებში** და გროვდება **შედეგების სპეციალურ ბუფერში**. პირობის გაანალიზების შედეგად თუ აღმოჩნდა, რომ გადასვლის მიმართულება სწორედ იყო ნაწინასწარმეტყველები, მაშინ ყველა მიღებული შედეგი ბუფერიდან გადაიწერება დანიშნულების ადგილზე და პროგრამის შესრულება გაგრძელდება ჩვეულებრივი თანამიმდევრობით. ნაწინასწარმეტყველები მიმართულება თუ არასწორი აღმოჩნდა, მაშინ ბუფერში არსებული შედეგები წაიშლება (ე. ი. ბუფერი გაიწმინდება) და გაუქმდება კონვეიერის მიერ ბრძანებების შესრულების შედეგად მიღებული შედეგები. კონვეიერში დაიწყება საჭირო განშტოებიდან ბრძანებების ჩატვირთვა.

ოპტიმალური მწარმოებლურობის უზრუნველყოფისათვის გადასვლები მაქსიმალურად ზუსტად უნდა იყოს ნაწინასწარმეტყველები. საწინააღმდეგო შემთხვევაში კონვეიერის მიერ ჩატარებული მუშაობა ფუჭი აღმოჩნდება. მაქსიმალური ეფექტურობისათვის ამორჩეული გადასვლებიდან **80%** მაინც უნდა იყოს სწორი. თანამედროვე პროცესორებისათვის იგი **95%**-ზე არანაკლებია.

**1993** წლის ბოლოს ფირმებმა **Apple Computer, IBM** და **Motorola** ერთობლვად დაამუშავა **RISC** არქიტექტურის სუპერსკალარული მიკროპროცესორი **Power PC**. **1994** წელს მისი გამოყენება დაიწყო კომპიუტერებში **Macintosh**. განვიხილოთ რა შესაძლებლობებს აძლევს **RISC** არქიტექტურა და სუპერსკალარულობა პროცესორს.

**სუპერსკალარული ეწოდება პროცესორს**, რომლისგამომთვლელ ბირთვში არის რამდენიმე ერთნაირი ფუნქციური კვანძი (ე. ი. რამდენიმე **ALU**, მათემატიკური თანაპროცესორი, მამრავლებელი ანუ **Integer multiplier**, ძვრის მოწყობილობა და ა. შ), რაც მას აძლევს

რადენიმე ინსტრუქციის (ბრძანების) ერთდროულად შესრულების უნარს, რასაც *ინსტრუქციების ღონეზე ინფორმაციის დამუშავების უნარი* ეწოდება. ინსტრუქციების ნაკადს გვემავს არა *სტატიკური კომპილატორი*, არამედ *დინამიკური გამოძივლელი ბირთვი*.

**RISC** (*Reduced Instruction Set Computer* – ბრძანების შემცირებული ნაკრებიანი კომპიუტერი) არქიტექტურა ეწოდება მიკროპროცესორის ისეთ არქიტექტურას, რომლის დროსაც ინსტრუქციების (ბრძანებების) გამარტივების წყალობით ადვილდება მათი დეკოდირება. გარდა ამისა, **RISC** არქიტექტურის დროს ყველა ბრძანება ერთნაირი სიგრძისაა, რაც აადვილებს მენსიერებიდან მათ ამორჩევას. დეკოდირებისა და მენსიიერებიდან ამორჩევის ზემოთ აღნიშნული ნოვაციები ამაღლებს მიკროპროცესორის სწრაფმოქმედებას.

სუპერსკალურობისა და, **RISC** არქიტექტურის წყალობით **Power PC** მიკროპროცესორი ერთ ტაქტში სამი ბრძანების შესრულებას ახერხებდა. ყველა წინა მიკროპროცესორი იყენებდა რთულ **CISC** (*Complex Instruction Set Computing* ანუ *Complex Instruction Set Computer*) ბრძანებებს, ან მიეკუთვნებოდა **RISC**-ბირთვის მქონე **CISC**-პროცესორების ჯგუფს; მათში გამოყენებული ბრძანებების სიგრძე **8**-დან **108** ბიტამდე ფარგლებში იცვლებოდა და ამიტომ ასეთ მიკროპროცესორებს უხდებოდა დამატებითი დრო დაეხარჯა ბრძანებების ამორჩევასა და დეკოდირებაზე.

**Power PC** პროცესორების ღირსება ისიც იყო, რომ ისინი (**603** და **604** მოდელებიდან დაწყებული) მოიხმარდა დაბალ სიმძლავრეს. ენერჯის დასაზოგად მათი ყველა დაუტვირთავი ელემენტი განირთვებოდა კვების წრედიდან და მას მხოლოდ საჭიროებისამებრ მიუერთდებოდა.

მიუხედავად ზემოთ ჩამოთვლილი ღირსებებისა, **2000** წლის დასაწყისში **Power PC** პროცესორის პლატფორმის განვითარება ჩიხში შევიდა. ახალი არქიტექტურის მიკროპროცესორების დამუშავება დიდ დროსა და ფულად დანახარჯებს მოითხოვდა, ამიტომ **2006** წელს **Apple** ფირმამ გადაწყვიტა კომპიუტერი **Macintosh** გადაეყვანა **Intel**-ის კომპიუტერებზე.

**2000** წლის მარტში **AMD** ფირმამ დაამზადა **Athlon K7** სახელწოდების მიოკროპროცესორი, რომლის ტაქტურმა სიხშირემ პი-



რველად გადააჭარბა *1* გიგაპეცს. ამან მიკროპროცესორული ტექნიკის ბაზარზე ფირმის პოზიცია გაამყარა.

**2003** წლის სექტემბერში **AMD** ფირმამ პირველად პერსონალური კომპიუტერებისათვის წარმოადგინა **64**-თანრივიანი მიკროპროცესორი **Athlon 64**.

მიკროცესორების სფეროში **2005** წლის ყველაზე მნიშვნელოვან მოვლენას წარმოადგენდა ბაზარზე **ორბირთვული ცენტრალური მიკროპროცესორის** გამოჩენა. **ბირთვი** მიკროპროცესორის ნაწილია, რომელიც შეიცავს ბრძანებათა ერთი ნაკადის შესასრულებლად საჭირო ყველა ფუნქციურ ბლოკს პირველი ორბირთვული მიკროპროცესორი იყო ფირმა **Intel**-ის მიერ გამოშვებული მიკროპროცესორი **Pentium D** და ფირმა **AMD**-ს მიერ გამოშვებული მიკროპროცესორი **Athlon 64X2**. ერთ-ერთ ყველაზე რევოლუციურ მრავალბირთვულ მიკროპროცესორად აღიარეს ფირმა **Intel**-ის მიერ **2006** წელს დამუშავებული მიკროპროცესორების ახალი მწკრივი **IntelCore 2 Duo**.

დღეისათვის ბაზარზე ძირითად მიკროპროცესორებად ითვლება **2**-დან **8**-მდე ბირთვის მქონე მრავალბირთვული მიკროპროცესორები. მათ შორის აღსანიშნავია **2011** წელს **Intel**-ის მიერ დამუშავებული ახალი თაობის ორბირთვული **IntelCore i3**, **ოთხბირთვული IntelCore i5** და **IntelCore i7**, აგრეთვე **AMD**-ს მიერ დამუშავებული ორბირთვული **AMD A4**, სამბირთვული **AMD A6**, ოთხბირთვული **AMD A8** და რვაბირთვული **AMD FX-8150** მიკროპროცესორები.

## 2.2. პერსონალური კომპიუტერების კლასიფიკაცია

კომპიუტერების კლასიფიკაციისაგან განსხვავებით არსებობს პერსონალური კომპიუტერების სპეციფიკური კლასიფიკაცია. პერსონალური კომპიუტერის მომხმარებელთა მოთხოვნების დასაკმაყოფილებლად, ნაკეთობის გასაიაფებლად, იმ პერიოდისათვის უახლოესი ოპერაციული სისტემა **Windows 98**-ის მხარდასაჭერად და ახალი ტექნოლოგიების დასანერგად **Intel** და **Microsoft** კორპორაციებმა დაამუშავა პერსონალური კომპიუტერების სპეციფიკაცია **PC'99 System Design Guide** („**PC'99** სისტემის დაპროექტების სახელმძღვა-

ნელო“, რომელმაც **1999-2000** წლებში განსაზღვრა პერსონალური კომპიუტერის სახე. ამ სპეციფიკაციის შესაბამისად პერსონალური კომპიუტერები შემდეგ ხუთ სახედ იქნა დაყოფილი:

**1. სამომხმარებლო პერსონალურ კომპიუტერებად (Consumer PC).** ისინი ლოკალური ქსელის გარეთ არსებულ, ოღონდ გლობალური ქსელის შემადგენლობაში შემაჯავალ **SONO** სექტორში სამუშაოდაა განკუთვნილი (**Small Office /Home Office – მცირე ოფისი/საოჯახო ოფისი**). ასეთი კატეგორიის კომპიუტერებს აქვს შედარებით მძლავრი გრაფიკული კონტროლერი და გამოიყენება სწავლებისათვის, გასართობად (თამაშებისათვის) და ა. შ. რეკომენდებულია მათში გამოიყენებული იყოს **Compaq, Intel** და **Microsoft** კომპანიების მიერ **1997** წელს დამუშავებული **Device Bay** სტანდარტი, რომელიც კორპუსის გაუნსნელად და კომპიუტერის გადაუტვირთავად პერიფერიის შეცვლის საშუალებას იძლევა.

**2. საოფისე პერსონალურ კომპიუტერებად (Office PC).** საოფისე პერსონალური კომპიუტერი სამომხმარებლო პერსონალური კომპიუტერისაგან ძირითადად განსხვავდება **ფლობის შედარებით დაბალი საერთო ღირებულებით**, გადაწერადი **BIOS**-ის არსებობით, შორიდან ჩატვირთვის შესაძლებლობითა და ლოკალურ გამოთვლით ქსელში მუშაობის შესაძლებლობით.

**ფლობის საერთო ღირებულება** (ინგლ. **Total Cost of Ownership, TCO**) ეწოდება იმ ღირებულებათა ჯამს, რომლებიც უნდა გაიღოს მფლობელმა გარკვეული ობიექტის დაუფლების მომენტიდან მფლობელობის უფლებიდან გამოსვლამდე; მასში შედის ფლობასთან დაკავშირებული ყველა ვალდებულების შესრულებისათვის გაწეული ხარჯებიც.

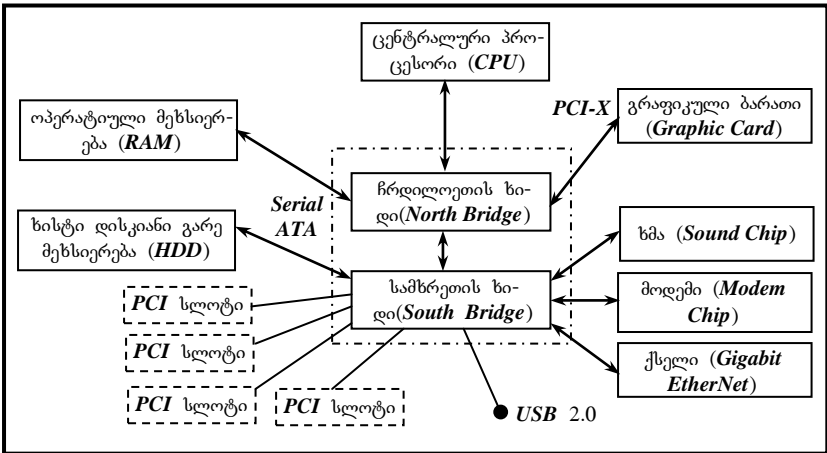
**3. გართობისათვის განკუთვნილი პერსონალური კომპიუტერებად (Entertainment PC).** მათ აქვს **Hi-Fi (High Fidelity – მაღალი სიზუსტე, მაღალი სანღობა)** კლასის აუდიოსისტემა, მთელ ეკრანზე **DVD** ფილმების ჩვენების უნარის მქონე მაღალმწარმოებლური სამგანზომილებიანი გრაფიკული კონტროლერი და ციფრული ტელეხედვა (**Digital TeleVision - DTV**); გარდა ამისა, ისინი შეიძლება **USB**-ს ან **Firewire**-ს (**USB – Universal Serial Bus – უნივერსალური მიმდევრობითი სალტე; Firewire – მაღალჩქაროსნული მიმდებრობითი სალტე**) მეშვეობით მივეუერთოთ სხვადასხვა მულტიმედიაურ მოწყობილობას – კომპოდერებს, ვიდეომაგნიტოფონებს და ა. შ. (**კომპოდერი ეწოდება კონსტრუქციულად გაერ-**

თანებულ მცირე გაბარიტიან სატელევიზიო კამერასა და კასეტურ ვიდეომაგნიტოფონს; ვიდეომაგნიტოფონი შეიძლება ხისტად იყოს ჩაშენებული კომპოდერში, ან შეიძლება მას ეხსნებოდეს).

**4. მობილურ პერსონალურ კომპიუტერებად (Mobile PC)**, რომლებიც გამოირჩევა მცირე გაბარიტებით, წონით, დაბალი მოხმარებული ელექტრული ენერჯითა და აკუმულატორის დაუმუხტავად ხანგრძლივად მუშაობით.

**5. მუშა სადგურებად (Workstation)**. ისინი განკუთვნილია რესურსტევადი ამოცანებისათვის, რომელთა გადასაწყვეტად საჭიროა ინტენსიური გამოთვლები (სამგანზომილებიანი მოდელირება, პროგრამული უზრუნველყოფის დამუშავება და ა. შ.).

თანამედროვე პერსონალური კომპიუტერები მნიშვნელოვნად განსხვავდება თავისი წინაპრებისაგან – დღეს მათში გამოიყენება უახლესი ტექნოლოგიები. კომპანია **Intel**-მა დაამუშავა «**მესამე თაობის შეტანა-გამოტანის**» კონცეფცია (**3GIO** ტექნოლოგია - «**Third Generation Input - Output**»), რომლის საფუძველზე შეიქმნა **PC'2003** სპეციფიკაცია (ნახ. 2.1).



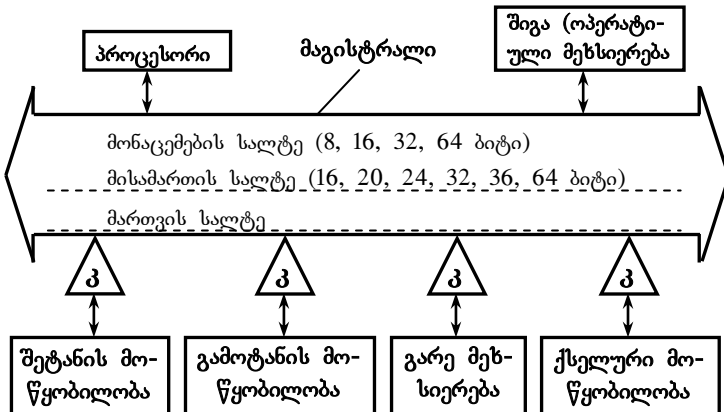
**ნახ.2.1 3GIO ტექნოლოგიის არქიტექტურა**

**3GIO** არქიტექტურაში შემოტანილია სიახლეები. მაგალითად, გრაფიკული კონტროლერისათვის გამოყენებული **AGP** სალტე იცვლება **PCI-X** ინტერფეისით. ასევე, მოსალოდნელია, რომ ხისტ დი-

სკებთან კავშირისათვის მოხდეს *Serial ATA*-სა და *USB 2.0*-ის ინტეგრირება. ამ სპეციფიკაციის რეალური დანერგვა **2005** წლიდან დაიწყო.

### 2.3. პერსონალური კომპიუტერების აგების მაგისტრალურ-მოდულური პრინციპი

კლასიკური გამომთვლელი მანქანის სტრუქტურის (ნახ. 1.3) თანახმად პერსონალური კომპიუტერი შედგება პროცესორისაგან, შიგა (ოპერატიული) მეხსიერებისაგან, გარე მეხსიერებისაგან, შეტანისა და გამოტანის მოწყობილობებისაგან. მათ ბოლო წლებში დამატა ქსელური მოწყობილობებიც (ნახ. 2.2).



ნახ. 2.2 პერსონალური კომპიუტერის აგების მაგისტრალურ-მოდულური არქიტექტურა

პროცესორს უხდება მონაცემები გაცვალოს მრავალრიცხოვან მოწყობილობებთან (შიგა მეხსიერებასთან, შეტანია მოწყობილობებთან, გამოტანის მოწყობილობებთან და ა. შ.). თითოეულთან დასაკავშირებლად პროცესორისათვის განცალკევებული არხის გამოყოფა შეუძლებელია. ამის ნაცვლად გამოიყენება კავშირის საერთო ხაზი, რომელიც თითოეულმა მოწყობილობამ შეიძლება რიგრიგობით გამოიყენოს. ასეთ საინფორმაციო არხს ეწოდება სალტე. **სალტე (ანუ მაგისტრალი)** წარმოადგენს პერსონალური კომპიუტერის რამდენიმე მოწყობილობას შორის მონაცემების გასაცვლელად განკუთვნილი

ხაზების ვგუფს. ტრადიციულად სალტე იყოფა შემდეგ სამ ნაწილად (იხ. ნახ. 2.2): მონაცემების სალტედ, მისამართის სალტედ და მართვის სალტედ. გავეცნოთ თითოეულ მათგანს.

**მონაცემების სალტე** ამ სალტით სხვადასხვა მოწყობილობას ერთმანეთს აწვდის მონაცემებს. მაგალითად, ოპერატიული მეხსიერებიდან წაკითხული მონაცემები შეიძლება დასამუშავებლად გადაეცეს პროცესორს, ხოლო უკვე დამუშავებული მონაცემები შესანახად გადაიგზავნოს ოპერატიულ მეხსიერებაში. ამგვარად, მონაცემების სალტით მონაცემები მოწყობილობიდან მოწყობილობას შეიძლება გადაეცეს ოპერატიული მეხსიერების გავლით.

მონაცემების სალტის თანრიგიანობა პროცესორის თანრიგიანობით, ე.ი. ორობითი თანრიგების იმ რაოდენობით განისაზღვრება, რომლებიც პროცესორმა შეიძლება ერთი ტაქტის განმავლობაში დაამუშაოს. პროცესორების თანრიგიანობა კომპიუტერული ტექნიკის განვითარების კვალბაზე იზრდება და დღეისათვის მისმა სიდიდემ **64** ბიტს მიაღწია.

**მისამართის სალტე** მოწყობილობას ან მეხსიერების უჯრედს, რომელშიც გადაიგზავნება, ან რომლიდან წაკითხება მონაცემები, ირჩევს პროცესორი. თითოეულ მოწყობილობას ან ოპერატიული მეხსიერების უჯრედს აქვს საკუთარი მისამართი. მისამართები გადაიგზავნება მისამართის სალტით, ამასთანავე ამ სალტით მისამართები გადაიცემა ერთი მიმართულებით – პროცესორიდან ოპერატიული მეხსიერებისაკენ ან მოწყობილობისაკენ. მაშასადამე, მისამართის სალტე ცალმხრივ მიმართული სალტეა.

მისამართის სალტის თანრიგიანობას განსაზღვრავს ოპერატიული მეხსიერების უჯრედების რაოდენობა, რომლებსაც შეიძლება ჰქონდეს უნიკალური მისამართები. მეხსიერების დამისამართებადი უჯრედების რაოდენობა შეიძლება გამოვიანგარიშოთ ფორმულით:  $N = 2^i$ , სადაც  $i$  არის მისამართის სალტის თანრიგიანობა.

მისამართის სალტის თანრიგიანობა მუდმივად იზრდება და **Pen-tium Edition** პროცესორების შემთხვევაში **64** ბიტს მიაღწია. ამგვარად, ასეთ პროცესორებში არსებობს  $N = 2^{64}$  რაოდენობის დამისამართებადი უჯრედები. მათი ერთობლობა წარმოქმნის **პროცესორის სამისამართო სივრცეს** და ეს სივრცე განსაზღვრავს მისამართის სალტის თანრიგიანობას.

**მართვის სალტი.** მართვის სალტით გადაიცემა მაგისტრალში ინფორმაციის გაცვლის ხასიათის განმსაზღვრელი სივანლები, რომლებსაც **მართვის სივანლები** ეწოდება. ისინი განსაზღვრავს მეხსიერებიდან წაკითხვის თუ მეხსიერებაში ჩაწერის ოპერაციაა შესასრულებელი, მათვე უნდა მოახდინოს მოწყობილობებს შორის ინფორმაციის სინქრონიზება

**მაგისტრალზე მიერთებული მოწყობილობების ურთიერთ-ზემოქმედება.** პროცესორმა მაგისტრალის მეშვეობით მონაცემები უნდა გაუცვალოს ოპერატიულ მეხსიერებასა და შეტანა-გამოტანის მოწყობილობებს. ეს პროცესი განვიხილოთ პროცესორიდან ოპერატიულ (შიგა) მეხსიერებაში მონაცემების ჩაწერის მაგალითზე.

მოცემული ამოცანის შესასრულებლად პროცესორს გამოაქვს:

- **მონაცემების სალტზე** - მეხსიერებაში ჩასაწერი მონაცემები;
- **მისამართის სალტზე** - ოპერატიული მეხსიერების მისამართი, რომელზეც უნდა ჩაიწეროს მონაცემები;
- **მართვის სალტზე** - ჩაწერის ოპერაციის ბრძანების კოდი.

აღნიშნული ოპერაციების შესრულების შემდეგ პროცესორი ელოდება სალტიდან მონაცემების „აღებას“. აღნიშნული ლოდინის პერიოდში ყველა დანარჩენი მოწყობილობა მუდმივად „უსმენს“ სალტეს (ამოწმებს მის მდგომარეობას).

მონაცემების სალტზე გამოტანილ მონაცემებს აღმოაჩენს **ოპერატიული მეხსიერება**. მართვის სალტზე არსებული ბრძანების ძალით მან იგი უნდა ჩაწეროს მისამართზე, რომელიც გამოტანილია მისამართების სალტზე. ეს დავალება მან უნდა შეასრულოს და შესრულების ფაქტი აცნობოს **პროცესორს**, ანუ, როგორც ამბობენ, **დაადასტუროს დავალების შესრულება**. მეხსიერების მუშაობის მაღალი საიმედოობის გამო ითვლება, რომ მის მიერ დავალების შესრულების ფაქტის გაკონტროლება უბრალოდ ფუჭად დაგვაკარგვინებს დროს და ამიტომ საჭირო არ არის: **პროცესორი** გარკვეული დროის გავლის შემდეგ პირდაპირ დაიწყებს შემდეგი ბრძანების შესრულებას. ამ მაგალითიდან ჩანს, რომ სალტით მონაცემების გასაცვლელად საჭიროა არსებობდეს ზუსტად განსაზღვრული წესები (მათ ერთობლიობას – **სალტის პროტოკოლი** ეწოდება), რომლებიც ყველა მოწყობილობამ უნდა დაიცვას.

ადრეული კომპიუტერებისაგან განსხვავებით, გარე მოწყობილობებთან პროცესორის ურთიერთობა დღეს სხვანაირადაა ორგანიზებული. კლასიკური არქიტექტურის დროს პროცესორი აკონტროლებდა შეტანა-გამოტანის ყველა პროცესს. ამის შედეგად სწრაფოქმედი პროცესორი გაცილებით ნელმოქმედი გარე მოწყობილობების ლოდინში ძალიან დიდ დროს კარგავდა. ამის გამოსარიცხად დამუშავდა კონტროლერებად წოდებული სპეციალური ელექტრონული სქემები, რომლებიც მართავდა პროცესორსა და გარე მოწყობილობებს შორის მონაცემების გაცვლის პროცესს. მესამე თაობის კომპიუტერებში მათ *შეტანა/გამოტანის არხები*, ხოლო მეოთხე თაობის კომპიუტერებში – *კონტროლერები* (ინგლ *Control* – ქართულად მართვას ნიშნავს). **2.2** ნახაზზე კონტროლერები ასო **ჟ**-თია აღნიშნული. *კონტროლერი* გარე მოწყობილობის მმართველი ელექტრონული სქემაა, რომელიც მონაცემების უმარტივესად ამუშავებს. განცალკევებული მიკროსქემის სახით დამზადებულ კონტროლერს – *მიკროკონტროლერი* ეწოდება.

მაგალითვისათვის განვიხილოთ გარე მეხსიერებად გამოყენებული თანამედროვე ხისტი დისკის კონტროლერი. მისი ძირითადი ამცანაა პროცესორიდან მიღებული კოორდინატების მიხედვით დიდკზე იპოვოს საჭირო მონაცემები, წაიკითხოს ისინი და გადასცეს ოპერატიულ მეხსიერებას. გარდა აღნიშნული ფუნქციისა, მას სხვა, ხშირად მეტად არატრივიალური ფუნქციების შესრულებაც შეუძლია. მაგალითად, იგი დისკის სამოსამსახურეო არეში შეინახავს ინფორმაციას მაგნიტურ ზედაპირზე არსებული ყველა უხარისხოდ დამზადებული სექტორის შესახებ (რომელთა გამორიცხვა მაღალი სიმჭიდროვით ჩაწერისას თანამედროვე ტექნოლოგიით ჯერჯერობით ვერ ხერხდება) და პროცესის მსვლელობაში მათ ცვლის სარზერვო სექტორებით, რაც დისკის უდევფექტობის ილუზიას გვიტოვებს.

პერსონალური კომპიუტერის *მაგისტალურ-მოდელური არქიტექტურის* (იხ. ნახ. **2.2**) ძალით მონაცემები გარე მოწყობილობებსა და ოპერატიულ (შიგა) მეხსიერებას შორის მონაცემები შეიძლება პირდაპირ, მიკროპროცესორის გარეშეც გაიცვალოს. გარდა ამისა, სალტეების არსებობის გამო მნიშვნელოვნად მარტივდება პერსონალურ კომპიუტერთან ახალი მოწყობილობების მიერთება, ე. ი. იგი შეიძლება ადვილად გაფართოვდეს. სალტის ლოგიკური და ფიზიკუ-

რი პარამეტრების სრულ ერთობლიობას ამ **სალტის სპეციფიკაცია-ეწოდება**. სალტის სპეციფიკაცია თუ ყველასათვის ცნობილია (საჯაროდაა გამოქვეყნებული), მაშინ სხვადასხვა მწარმოებელს შეუძლიათ ამ სალტისათვის დაამუშაოს ნევბისმიერი დამატებითი მოწყობილობა. ასეთ მიდგომას **ღია არქიტექტურის პრინციპი** ეწოდება. ამ დროს კომპიუტერში გათვალისწინებულია სტანდარტული ახალი მოწყობილობების მასთან მისაერთებელი სტანდარტული გასართები. ეს მომხმარებელს საშუალებას აძლევს, ააწყოს მისთვის საჭირო ფუნქციების შემსრულებელი პერსონალური კომპიუტერი; ამ დროს უნდა გვანსოვდეს, რომ ნებისმიერი ახალი მოწყობილობის მისაერთებლად საჭიროა კომპიუტერში ჩავტვირთოთ ამ მოწყობილობასა და პროცესორს შორის მონაცემების გაცვლის პროცესის მმართველი სპეციალური პროგრამა – **დრაივერი**.

თანამედროვე კომპიუტერებში მუშაობის ეფექტურობის ასამაღლებლად რამდენიმე სალტე გამოიყენება; მაგალითად, ერთ-ერთი პროცესორს აკავშირებს მეხსიერებასთან, მეორე – პროცესორს – ვიდეოსისტემასთან. ასეთ სალტეებს უფრო ლეტალურად მიმდევრო პარაგრაფში განვიხილავთ.

არსებობს ცენტრალურ პროცესორსა და გარე მოწყობილობებს შორის მონაცემების გაცვლის შემდეგი სამი რეჟიმი: **1.** პროგრამულად მართული შეტანა/გამოტანა; **2.** შეწყვეტებით მონაცემების გაცვლა; **3.** მეხსიერებასთან პირდაპირი დაშვება.

**პროგრამულად მართული გაცვლის** დროს შეტანა-გამოტანისათვის საჭირო ყველა ქმედება პროგრამითაა გათვალისწინებული. პერიფერიული მოწყობილობის მზადყოფნის მდგომარეობაში გადასვლისა და სხვა დროებითი შეყოვნებების ჩათვლით გაცვლის პროცესს მთლიანად პროცესორი მართავს. ამ მეთოდის ღირსებაა სიმარტივე და ის, რომ არავითარი დამატებითი მოწყობილობა საჭირო არ არის; ნაკლია – დროის დიდი დანაკარგი, რაც იმითაა განპირობებული, რომ შეტანა/გამოტანის ნელმოქმედი მოწყობილობები დროს აკარგვენებს სწრაფმოქმედ პროცესორს.

**შეწყვეტებით გაცვლის დროს** შეტანა-გამოტანის მოწყობილობები საჭიროებისამებრ თვითონვე „მიიპყრობს“ პროცესორის ყურადღებას. მაგალითად, კლავიშზე თითის ყოველ დაჭერას, ან მისგან თითის აშვებას კლავიატურა ატყობინებს პროცესორს; დანარჩენი დროის



განმავლობაში პროცესორი კლავიატურისათვის ყურადღების მიუქცევლად ასრულებს თავის სამუშაოს. კლავიშზე ზემოთ აღნიშნული ყოველი ზემოქმედებისას ცენტრალური პროცესორი გარკვეული დროის განმავლობაში „გვერდზე გადადებს“ ძირითადი პროგრამის შესრულებას და გადადის შეწყვეტის დამუშავების სამომსახურეო პროგრამაზე. მისი დამთავრების შემდეგ ცენტრალური პროცესორი უბრუნდება პროგრამის იმ წერტილს, რომელზეც იგი შეწყვეტეს.

წარმოვიდგინოთ, რომ კათედრის სხდომის დროს ტელეფონით შემოვიდა სასწრაფოდ შესასრულებადი მნიშვნელოვანი ინფორმაცია. კათედრის ტექნიკური პესონალი სხდომის დამთავრების დაულოდებლად ატყობინებს კათედრის გამგეს ზარის შესახებ. იგი წყვეტს სხდომას, ტელეფონით ერკვევა საქმის არსში, ატყობინებს თავის გადაწყვეტილებას და შემდეგ განაგრძობს სხდომას. აქ ცენტრალური პროცესორის როლს ასრულებს კათედრის გამგე, ხოლო ტელეფონისას – *შეწყვეტის მოთხოვნა*. ზემოთ აღნიშნული ტექნიკური პერსონალის კომპიუტერული ანალოგია *შეწყვეტების კონტროლერი*, რომელიც შემოსულ „ზარებს“ (*შეწყვეტებს*) მათი მნიშვნელობების (*პრიორიტეტების*) შესაბამისად ახარისხებს.

შეწყვეტების მექანიზმი არა მარტო აპარატურულ ნაწილში, არამედ გარკვეული მოვლენების (კლავიშებზე დაჭერის, მაუსის მართვის ბრძანებების და ა. შ.) დამამუშავებელ პროგრამებშიც გამოიყენება. ასეთი ტექნოლოგია უდევს საფუძვლად ყველა თანამედროვე ოპერაციულ სისტემებსა და გამოიყენება *Microsoft Visual Studio, Delphi, Lazarus* და მსგავსი პროგრამების დამუშავებისას.

ზემოთ აღწერილ ორივე ვარიანტში მონაცემების გაცვლას მართავს პროცესორი. მეხსიერებიდან მას ამოაქვს, ან მეხსიერებაში იგი ჩაწერს მონაცემებს, პროცესორივე აკონტროლებს სალტების მუშაობას და ასრულებს საჭირო დამუშავებას. გადასაცემი მონაცემები თუ რთულ დამუშავებას არ საჭიროებს, მაშინ არ ღირს მონაცემების გასაცვლელად ცენტრალური პროცესორის მოცდენა. ამ სამუშაოსაგან პროცესორის განსათავისუფლებად და სხვადასხვა მიმართულებით (მოწყობილობიდან მეხსიერებაში ან მეხსიერებიდან – მოწყობილობაში) მონაცემების მსხვილი ბლოკების სახით გადასაცემად გამოიყენება *მეხსიერებასთან პირდაპირი დაშვება*, ანუ *DMA (Direct Memory Access)*. მისთვის დამახასიათებელ ძირითად თავისე-

ბურებაა ის, რომ პროცესორი მონაცემებს კი არ გადაცემს, არამედ იგი გადაცემას ამზადებს მეხსიერებასთან პირდაპირი დაშვების კონტროლერის დაპროგრამების გზით; კერძოდ, ამყარებს გაცვლის რეჟიმს და ოპერატიულ მეხსიერებას ატყობინებს გაცვლის ციკლების საწყის მისამართსა და გადასაცემი ციკლების რაოდენობას.

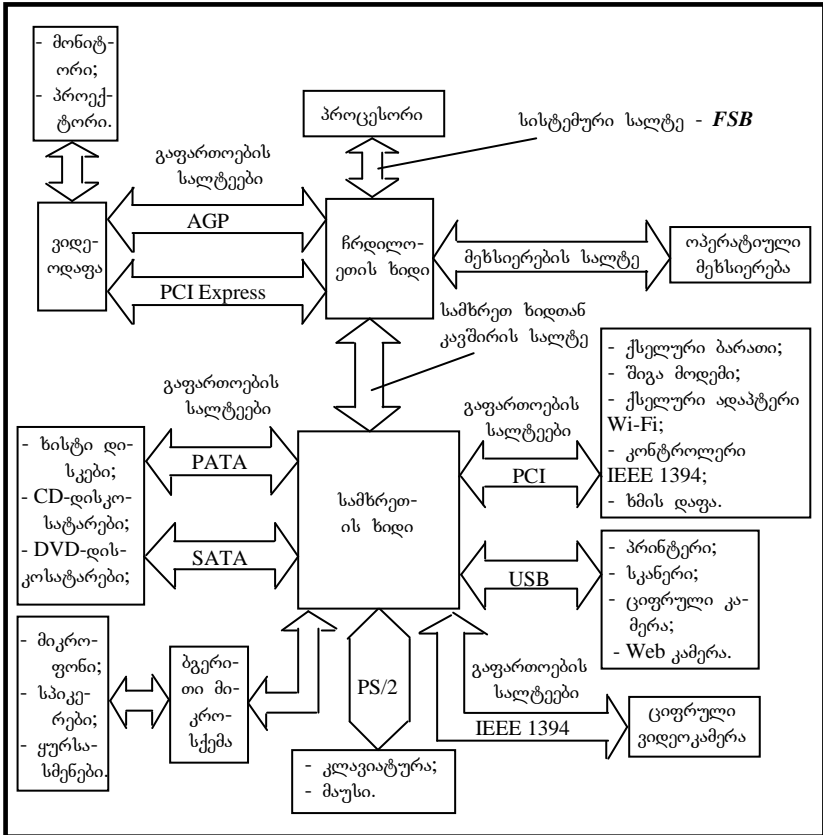
## 2.4. სისტემური ღაფა (დედა-ღაფა)

**სისტემური ღაფა** პერსონალური კომპუტერის უმნიშვნელოვანესი აპარატურული კომპონენტია (*ნახ. 2.3*). მას ხშირად **დედა-ღაფა-საცუწოდებენ**. სისტემურ ღაფაზეა რეალიზებული ინფორმაციის გაცვლისათვის განკუთვნილი მაგისტრალი; მასზევეა განთავსებული პროცესორის დასაყენებელი **გასართები**, აგრეთვე სლოტები, რომლებზედაც ყენდება ოპერატიული მეხსიერება და სხვადასხვა კონტროლერი. შევნიშნავთ, რომ **სლოტი** ეწოდება ხვრელურ გასართს.

პროცესორის, სისტემური სალტებისა და პერიფერიული მოწყობების სწრაფმოქმედებები ერთმანეთისაგან განსხვავდება. ზოგადად **მოწყობილობის სწრაფმოქმედება** დამოკიდებულია ტაქტური გენერატორის ტაქტურ სიხშირესა (*ჩვეულებრივ იგი მეგაჰერცით იზომება, ხოლო 1 ჰერცი წამში ერთი რხევის ტოლია; 1 ჰც = 1/წმ = 1 წმ<sup>-1</sup>*) და ამ მოწყობილობის თანრიგიანობაზე, ე.ი. მონაცემების იმ რაოდენობის ბიტებზე, რომელთა ერთდროულად დამუშავება ან გადაცემა შეუძლია ამ მოწყობილობას (იზომება ბიტებით). სიხშირე მოწყობილობის შიგნით დამატებით შეიძლება სხვადასხვა კოეფიციენტზეც გადამრავლდეს. შესაბამისად, სხვადასხვა მოწყობილობების შემადგენელ სალტებს განსხვავებული სიჩქარით უნდა შეეძლოს მონაცემების გადაცემა, ე.ი. მათ გატარების სხვადასხვა უნარი უნდა ჰქონდეს.

**გატარების უნარი** მოწყობილობის სისწრაფე დამოკიდებულია ტაქტური გენერატორის ტაქტურ სიხშირესა (*ჩვეულებრივად იზომება მეგაჰერცებით - მჰც*) და თანრიგიანობაზე, ე.ი. იმ მონაცემთა ბიტების რაოდენობაზე, რომლებიც შეუძლია მოწყობილობას ერთდროულად დაამუშაოს ან გადასცეს (იზომება ბიტებით). მოწყობილობებში სიხშირეები დამატებით სხვადასხვა კოეფიციენტზე მრავლდება.

ზემოთ აღნიშნულის თანახმად, მაგისტრალზე მიერთებულ სხვადასხვა მოწყობილობების სწრაფმოქმედებები სხვადასხვაა. შესაბამისად ამ მოწყობილობების შემაერთებელ სხვადასხვა სალტეშიც მონაცემები სხვადასხვა სიჩქარით უნდა გადაიცეს



ნახ. 2.3 სისტემური დაფის (დედა-დაფის) ლოგიკური სქემა

მონაცემების სალტის გატარების  $H$  უნარი (იზომება ბიტი/წმ-ით) უდრის სალტის  $Q$  თანრივიანობისა (იზომება ბიტობით) და სალტის  $W$  სიხშირის (იზომება გიგაჰერცებით -**გჰც**) ნამრავლს:

$$H = QW. \tag{2.1}$$

**ჩრდილოეთისა და სამხრეთის ხიდები.** ტაქტური სიხშირისა და მოწყობილობების თანრიგთანობის ურთიერთშესათანხმებლად სისტემურ დაფაზე ყენდება სპეციალური მიკროსქემები (მათ ნაკრებს **ჩიპსები** ეწოდება), რომლებიც მოიცავს ოპერატიული მეხსიერებისა და ვიდეომეხსიერების კონტროლერს (ე. წ. **ჩრდილოეთის ხიდს**) და პერიფერიული მოწყობილობების კონტროლერს (**სამხრეთისხიდს**).

**პროცესორის სიხშირე.** ჩრდილოეთის ხიდი უზრუნველყოფს პროცესორს, ოპერატიულ მეხსიერებასა და ვიდეომეხსიერებას შორის მონაცემების გაცვლას. პროცესორის სიხშირე რამდენჯერმე აღემატება სისტემური სალტის საბაზისო სიხშირეს. **სისტემურ სალტეს** სმირად **FSB (Front Side Bus)** სალტესაც უწოდებენ (იხ. ნახ. 2.3). მაგალითად, **2006** წელს დამზადებულ ყველაზე სწრაფ კომპიუტერებში **FSB** სალტის სიხშირე **266** მგჰც-ის ტოლი, ხოლო გამრავლების კოეფიციენტი **14**-ის ტოლი იყო. მაშასადამე, პროცესორის სიხშირე უდრიდა  $266 \text{ მგჰც} \times 14 \approx 3,7$  გიგაჰერცს.

**სისტემური სალტი.** ჩრდილოეთის ხიდსა და პროცესორს შორის მონაცემები **FSB** სალტის სიხშირეზე **4**-ჯერ მეტი სიხშირით გადაიცემა ამიტომ ზემოთ აღნიშნული კომპიუტერის შემთხვევაში პროცესორს მონაცემები გადაეცემა  $4 \times 266 = 1054 \text{ მგჰც}$  სიხშირით. რადგან სისტემური სალტის თანრიგების რაოდენობა ემთხვევა პროცესორის თანრიგების რაოდენობას და მოცემულ შემთხვევაში **64** ბიტის ტოლია, ამიტომ სისტემური სალტის გატარების უნარი იქნება:  $H_{FSB} = 64 \text{ ბიტი} \times 1054 \text{ მგჰც} \approx 8 \text{ გბაიტი/წმ}$ .

**მეხსიერების სალტი** გამოიყენება ჩრდილოეთის ხიდის გავლით პროცესორსა და ოპერატიულ მეხსიერებას შორის მონაცემების გადასაცემად (იხ. ნახ. 2.3). ამ სალტის თანრიგების რაოდენობა ემთხვევა სისტემური სალტის თანრიგების რაოდენობას, მაშასადამე, ზემოთ განხილული კომპიუტერის სისტემური დაფაზე არსებული მეხსიერების სალტეს აქვს **64** თანრიგი

მეხსიერების სალტის სიხშირე ჩამოუვარდება სისტემური სალტის სიხშირეს. მაგალითად, მეხსიერების სალტის სიხშირე შეიძლება **533 მგჰც**-ის ტოლი იყოს, ანუ ოპერატიული მეხსიერება ორჯერ უფრო იშვიათად იღებდეს მონაცემებს ვიდრე პროცესორი. ამი-

ტომ მოცემული კომპიუტერისათვის მეხსიერების სალტის გატარების უნარი იქნება:

$$H_{\text{მეხ.სალტე}} = 64 \text{ ბიტი} \times 533 \text{ მკვ} \approx 4 \text{ მბაიტი/წმ.}$$

შეენიშნავთ, რომ ბოლო წლებში დამუშავდა პროცესორები, რომლებიც ოპერატიულ მეხსიერებას ჩრდილოეთის ხიდის გარეშე პირდაპირ უკავშირდება.

**გაფართოების სალტეები** ეწოდება სალტეებს, რომლებითაც სისტემურ დაფას შეიძლება მიეერთოთ პერიფერიული მოწყობილობები. **2.3** ნახაზის თანახმად გვაქვს გაფართოების შემდეგი სახის სალტეები:

**▲AGP და PCI express სალტეები.** დასამუშავებელი გრაფიკული ინფორმაციის გართულების კვალობაზე უნდა გაიზარდოს პროცესორთან და ოპერატიულ მეხსიერებასთან ვიდუო მეხსიერების დამაკავშირებელი სალტის სწრაფმოქმედებაც. ჩრდილოეთის ხიდთან ვიდუოდაფის მისაერთებლად შეიძლება გამოვიყენოთ **32-ბიტური AGP სალტე (Accelerated Graphic Port** – დაჩქარებული გრაფიკული პორტი). ეს სალტე თავდაპირველად მონაცემებს **66 მკვ** სიხშირით, ხოლო ახლა **528 მკვ** სიხშირით გადასცემს. ამიტომ მისი გატარების უნარია:

$$H_{AGP} = 32 \text{ ბიტი} \times 528 \text{ მკვ} \approx 2 \text{ მბაიტი/წმ.}$$

დღეისთვის სამხრეთ ხიდთან ვიდუოდაფის მისაერთებლად სულ უფრო და უფრო ხშირად გამოიყენება **PCI express სალტე (Peripheral Component Interconnect bis Express** – პერიფერიული მოწყობილობების ურთიერთმოქმედების აჩქარებული სალტე). ამ სალტის გატარების უნარი მნიშვნელოვნად აღემატება **PCI** და **AGP** სალტეების **გატარების უნარს**.

ვიდუოდაფას ანალოგური **VGA (Video Graphic Array** – გრაფიკული ვიდუოდაპტერი) ან ციფრული **DVI (Digital Vizual Interface** – ციფრული ვიდუოინტერფეისი) გასართის დახმარებით შეგვიძლია მიეუერთოთ ელექტრონულსხივური ან თხევად-კრისტალური მონიტორი ან პროექტორი.

**▲PCI სალტე.** ჩრდილოეთის ხიდს სპეციალური სალტით უკავშირდება სამხრეთის ხიდი, რომელთანაც პერიფერიული მოწყობილობებია მიერთებული. **PCI (Peripheral Component Interconnect bus** – პერიფერიული მოწყობილობებთან ურთიერთზემოქმედების სალტე) სალტე უზ-

რუნველყოფს ინფორმაციის გაცვლას სისტემური დაფის გაფართოების სლოტებში დაყენებულ პერიფერიულ მოწყობილობებთან.

ყველაზე ხშირად ეს სალტე გამოიყენება ლოკალურ ქსელში (ქსელური ბარათი), გლობალურ ქსელ ინტერნეტში (ჩაშენებული მოდემი) და უსადენო ქსელში (ქსელური **Wi-Fi** ადაპტერი) ჩასართველი მოწყობილობებისათვის.

**PCI** სალტის თანრიგთანობა შეიძლება იყოს **32** ან **64** ბიტის, ხოლო სიხშირე – **33** მჰც-ს ან **66** მჰც-ს ტოლი. მაშასადამე, **PCI** სალტის მაქსიმალური გამტარობის უნარია:

$$H_{PCI} = 64 \text{ ბიტი} \times 64 \text{ მჰც} \approx 528 \text{ მბაიტი/წმ.}$$

▲ **IEEE 1394 სალტე.** მას **FireWire**, **i-Link** სალტის სახელითაც მოიხსენებენ. იგი მიმდევრობითი მაღალჩქაროსნული სალტეა, რომელიც გამოიყენება კომპიუტერსა და ციფრულ მოწყობილობებს (ციფრულ ვიდეოკამერებს, **DVD**-პლეერებსა და ა. შ.) შორის ინფორმაციის გასაცვლელად გამოსახულებისა და ხმის ხარისხის გაუარესების გარეშე. ამ სალტის გამტარობის უნარმა შეიძლება მიაღწიოს ან გადააჭარბოს **200 მბაიტი/წმ-ს**.

▲ **ATA სალტე.** გარე მეხსიერების მოწყობილობები (ხისტი დისკები, **CD**- და **DVD**-დისკოსატარები) სამხრეთის ხიდს **ATA** სალტის (**Advanced Technology Attachment** – დამგროვებლების მისაერთებელი სალტე) საშუალებით უერთდება. ადრე გამოიყენებოდა პარალელური **PATA** სალტე, რომლითაც მონაცემების გადაცემის სიჩქარეს შეიძლებოდა **133 მბაიტი/წმ** სიდიდისათვის მიეღწია. დღეს გავრცელდა მიმდევრობითი **SATA** სალტე, რომელშიც მონაცემების სიჩქარე შეიძლება **300 მბაიტი/წმ** სიდიდის ტოლია.

▲ **USB სალტე.** სამხრეთ ხიდის მეშვეობით კომპიუტერთან პრინტერების, სკანერების, ციფრული კამერების და სხვა პერიფერიული მოწყობილობის მისაერთებლად გამოიყენება **USB (Universal Serial Bus** - **უნივერსალური მიმდევრობითი სალტე)** სალტე. მისი გამტარობის უნარია **60 მბაიტი/წმ** და უზრუნველყოფს კომპიუტერთან ერთდროულად რამდენიმე პერიფერიული მოწყობილობის (პრინტერის, სკანერის, ციფრული კამერის, კამერის, მოდემის და ა.შ.) მიერთებას.

▲ **კლავიატურა და მუსი** კომპიუტერს უერთდება **PS/2** პორტის ან **USB** სალტის (მათ შორის უმავთულო ადაპტერის) მეშვეობით.

სამხრეთ ხიდს შეიძლება მიუერთდეს სისტემურ დაფაში ინტეგრირებული მიკროსქემა, რომელიც უზრუნველყოფს ხმის დამუშავებას (ეს ფუნქცია შეიძლება შეასრულოს *PCI* სალტით მისაერთებელმა ხმის დაფამ). აუდიოგასართების დახმარებით სისტემურ დაფას შეგვიძლია მივაერთოთ მიკროფონი, სპიკერები ან ყურსასმენები.

## 2.5. მიკროპროცესორი: პირითადი ელემენტები და მახასიათებლები

კომპიუტერის უმნიშვნელოვანეს მოწყობილობაა ცენტრალური პროცესორი (*CPU* – „*C*entral *P*rocessing *U*nit“ – „ცენტრალური გამოთვლელი მოწყობილობა“). სწორედ პროცესორის ტიპსა და მახასიათებლებზეა, უპირველეს ყოვლისა, დამოკიდებული მთლიანად კომპიუტერული სისტემის მწარმოებლურობა.

*ცენტრალური პროცესორი* ეწოდება მოწყობილობას, რომლის დანიშნულებაა მონაცემებზე შეასრულოს არითმეტიკულ/ლოგიკური ოპერაციები და უზრუნველყოს კომპიუტერის ყველა მოწყობილობის კოორდინირებული მუშაობა.

### დამზადების ტექნოლოგია

ფიზიკურად პროცესორი წარმოადგენს დიდ ნახევარგამტარულ მიკროსქემას. პროცესორები მზადდება სუფთა კაჟბადის (სილიციუმის)

ზოლიდან ამონაჭერი *ფუძეშრების* საფუძველზე. ტერმინი „*ფუძეშრე*“ მასალათმცოდნეობაში აღნიშნავს ძირითადი მასალის ზედაპირს, რომელიც სხვადასხვა სახის დამუშავებისათვის არის განკუთვნილი. *მიკროელექტრონიკაში* იგი ჩვეულებრივ მონოკრისტალური ნახევარგამტარული ფირფიტაა, რომელზეც სპეციალური ტექნოლოგიის გამოყენებით წარმოიქმნება ჰეტეროსტრუქტურები (სხვადასხვა ნახევარგამტარებისაგან წარმოშობილი შრისებური სტრუქტურები) ან წამოიზრდება მონოკრისტალური შრეები. კაჟბადი ნახევარგამტარია, რომელიც სხვადასხვა პირობებში იქცევა ელექტრული დენის გამტარავით ან იზოლატორივით.

თავდაპირველად კაჟბადის ფუძეშრეზე მაღალი ტემპერატურისა და ჟანგბადის ზემოქმედებით ფორმირდება *კაჟბადის დიოქსიდის შრე*. ეს პროცესი ძალიან ჰგავს წყალში ჩადებულ რკინაზე ჟანგის

წარმოქმნის პროცესს, ოღონდ ფუძემრეზე კაჟბადის დიოქსიდის შრე გაცილებით სწრაფად ფორმირდება და იგი იმდენად თხელია, რომ შეუიარაღებელი თვალით არ ჩანს.

ამის შემდეგ გამოიყენება ფოტოლითოგრაფიის მეთოდი. **ფოტოლითოგრაფია** გარკვეული მასალის ზედაპირზე გრაფიკული სურათის მიღების მეთოდია, რომელიც წარმატებით გამოიყენება სხვადასხვა სახის ინტეგრალური სქემაში. კერძოდ, ფუძემრეზე მიღებული კაჟბადის დიოქსიდის შრე დაიფარება ფოტომგრძობიარე პოლიმერული ფირით (**ფოტორეზისტორით**), რომელსაც **ფოტოშრე** ვუწოდოთ. ამ ფოტოშრეზე უნდა მოვახდინოთ საჭირო სქემის პროექცირება. ამისათვის ეს სქემა უნდა დავხაზოთ **შაბლონის** ფუნქციის შემსრულებელ სპეციალურ ნიღაბზე. შაბლონი უნდა მოვათავსოთ ფოტოშრის ზემოთ და დავამუქოთ ულტრაიისფერი სხივებით. ფოტოფირის მსგავსად ფოტოშრეზე დაიტანება პროექცირებული სქემის სურათი, რომელიც **სამჭავენებლით** უნდა გამოვამჟღავნოთ. სამჭავენებელი ფოტოშრისა და კაჟბადის დიოქსიდის დასხივებულ უბნებს მთლიანად „ამოჭამს“. ამის შემდეგ ფუძემრე უნდა „დავბომბოთ“ სხვადასხვა მინარევების იონებით. „ამოჭმული“ უბნების გავლით ფუძემრეში მოხდება „დამბომბველი“ იონების იმპლანტირება და ამ იმპლანტანტებით მასზე ფორმირდება (მასში „ჩაშენდება“) შაბლონზე არსებული სქემის ანალოგური ელექტროტრონული მიკროსქემა.

ზემოთ აღწერილი პროცესი რამდენჯერმე უნდა გავიმეოროთ და შევქმნათ რამდენიმე ათეული მიკროსქემა, რომელთა შორის უნდა დავტოვოთ ღიობები. ისინი საჭიროა მიკროსქემების ურთიერთშემადგერთებელი ხაზების წარმოსაქმნელად, რისთვისაც საკმარისია ღიობები ამოვავსოთ ლითონის (ალუმინის ან სპილენძის) ატომებით. მიკროსქემების ურთიერთშეერთებით წარმოიქმნება მიკროპროცესორის შესაბამისი **რთულ სამეანზომილებიანი ელექტრული მიკროსქემა**, რომელსაც **ინტეგრალური** (ანუ განუყოფლად დაკავშირებული მიკროსქემებისაგან წარმოქმნილი) **სქემა** ეწოდება.

ბოლო ეტაპზე მიკროპროცესორის წარმომქმნელი ინტეგრალური სქემა ჩაშენდება სისტემურ დაფასთან მიკროპროცესორის ელექტრულად მისაერთებელ დამცავ კორპუსში (კორპუსთან კაჟბადური

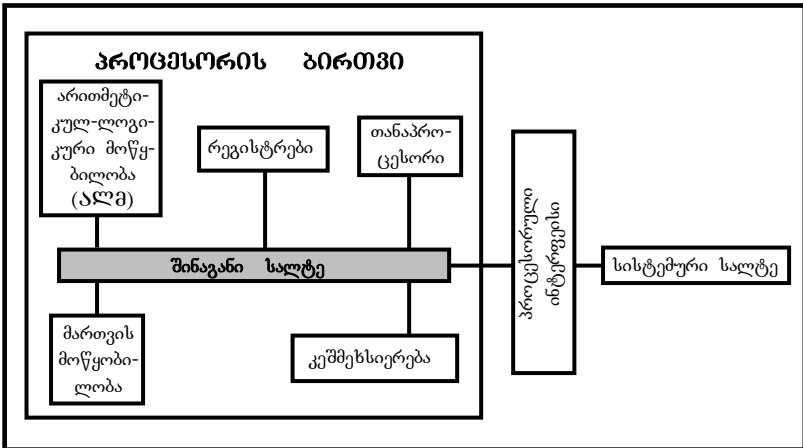


მიკროსქემის დასაკავშირებლად გამოიყენება ოქროსგან დამზადებული გამტარები).

1971 წელს დამზადებულ **i4004** ტიპის პირველ მიკროპროცესორში გამოიყენებოდა **2300** ტრანზისტორი და თითოეული მათგანის ზომა **10** მიკრონის ( $10^{-5}$  მეტრის) ტოლი იყო, რაც ადამიანის თმის ზომას შეიძლება შევადაროთ. თანამედროვე მიკროპროცესორების უმრავლესობაში თითოეული ელემენტის (ტრანზისტორის) ზომა **65** ნანომეტრის ტოლია ( $65$  ნანომეტრი =  $6,5 \cdot 10^{-8}$  მეტრს, რაც შეიძლება რამდენიმე ათეული ატომის ზომას შევადაროთ), ხოლო მათი რაოდენობა რამდენიმე ასეული ათასიდან მილიარდის ფარგლებში იცვლება.

**მიკროპროცესორის აგებულება**

მიკროპროცესორის აგებულების გამარტივებული სქემა **2.4** ნახაზზეა მოყვანილი. მისი ძირითადი ელემენტია **ბირთვი**, რომელიც პროცესორის ბევრი მახასიათებელს განსაზღვრავს. **ბირთვი** ეწოდება მიკროპროცესორის ყველა ძირითადი ფუნქციური ბლოკების შემცველ და ბრძანებათა ცალკეული ნაკადების შემსრულებელ ნაწილს.



**ნახ.2.4.** მიკროპროცესორის აგებულების ბლოკური სქემა

ადრეული მიკროპროცესორები შეიცავდა ერთადერთ ბირთვს. ასეთი მიკროპროცესორის მწარმოებლობის ამაღლების ერთადერთი გზა იყო ტაქტური სიხშირის გაზრდა, რაც ზრდიდა მიკროპროცესორის მიერ გამოყოფილ სითბოსა და მოხმარებულ ელექტროენერ-

გაიას. გამოყოფილი სითბოსა და მოხმარებული ელექტროენერჯის სი-  
დიდეებმა ისეთ ზღვრულ მნიშვნელობებს მიაღწია, რომ ტექტური  
სიხშირის შემდგომი ამალღება მიუღებელი გახდა. გამოსასვლელის  
პოვნის პროცესში დაიბადა ერთი ბირთვის ნაცვლად რამდენიმე ბი-  
რთვის გამოყენების იდეა. იგი წარმატებით იქნა რეალიზებული და  
გამოთვლითი ტექნიკის ბაზარზე გამოჩნდა **მრავალბირთვული მი-  
კროპროცესორები**. რამდენიმე ბირთვი მათში მონაცემებს პარალე-  
ლურად ამუშავებს რაც ზრდის მიკროპროცესორის მწარმოებლუ-  
რობას.

მიკროპროცესორის ბირთვი თავსდება პლასტმასის ან კერამიკის  
კორპუსში და სადენებით უერთდება ლითონურ გამომყვანებს (ფე-  
ხებს), რომლებითაც პროცესორი სისტემურ დაფასთანაა დაკავ-  
შირებული. მოკლედ განვიხილოთ მიკროპროცესორში არსებული  
კვანძები (იხ. ნახ. 2.4).

**1. არითმეტიკულ-ლოგიკური მოწყობილობა(ალმ)** უმარტივე-  
ეს შეთხვევაში შედგება ორი რეგისტრისა და ოპერაციების მმართ-  
ველი სქემისაგან. ოპერაციის შესრულებისას რეგისტრებში თავსდება  
საწყისი მონაცემები, რომლებზეც სუმატორი ასრულებს შეკრების  
ოპერაციას. ყველა დანარჩენი არითმეტიკული (გამოკლების, გამრავ-  
ლების, გაყოფის) ოპერაციები ამა თუ იმ ხერხით დაიყვანება შეკ-  
რების ოპერაციაზე. გამრავლებისა და გაყოფის ოპერაციების სწრა-  
ფად შესასრულებლად არც თუ ისე იშვიათად კონსტრუქტორები  
ართულებენ **ალმ**-ს. მაგალითად, პროცესორებში ფართოდ გამოიყე-  
ნება მცირე რიცხვების მზა ნამრავლების შემცველი ცხრილების მე-  
შვეობით რიცხვების გამრავლების მეთოდი.

არითმეტიკულ-ლოგიკური მოწყობილობა არა მატო გამოთვლებს  
ასრულებს, არამედ აანალიზებს გამოთვლების შედეგებსაც. ჩვეულ-  
ებრივ მოწმდება ორი პირობა: შედეგი ტოლია თუ არა ნულის (შე-  
დეგის ყველა თანრიგის ნულზე ტოლობა) და ან იგი უარყოფითია  
თუ არა (ნიშნის თანრიგის 1-ზე ტოლობა). ამ ანალიზის შედეგი  
შეიტანება პროცესორის **მდგომარეობის რეგისტრში** და მისი გა-  
მოყენებით განისაზღვრება შემდეგი პირობების ჭეშმარიტობა და  
მცდარობა:  $R = 0$ ;  $R \neq 0$ ;  $R > 0$ ;  $R < 0$ ;  $R \leq 0$ ,  $R \geq 0$ , სადაც  $R$ -ით  
აღნიშნულია ოპერაციის შედეგი. ეს პროგრამაში განშტოების ორგა-  
ნიზების საშუალებას გვაძლევს; მაგალითად, არაუარყოფითი რი-

ცხვისთვის შეიძლება შესრულდეს კვადრატული ამოფესვის ოპერაცია, ხოლო უარყოფითი რიცხვის დროს გაიცეს შეცდომის წარმოშობის შეტყობინება.

ართმეტიკულ ლოგიკური მოწყობილობა ოპერაციებს, როგორც წესი, მხოლოდ მთელ რიცხვებზე ასრულებს; ნამდვილ რიცხვებზე ოპერაციების ჩასატარებლად გამოიყენება თანამედროვე მიკროპროცესორის შიგნით ჩაშენებული **მათემატიკური თანაპროცესორი**

**2. მართვის მოწყობილობა.** მართვის მოწყობილობის მთავარი ამოცანაა პროცესორის მუშაობის ძირითადი ალგორითმის შესაბამისად უზრუნველყოს პროგრამის მიმდევრობის ავტომატური შესრულება. იგი ასრულებს შემდეგ ფუნქციებს:

- მენსიერებიდან ამოაქვს მომდევნი ბრძანება;
- ბრძანების გაშიფვრის გზით განსაზღვრავს შესასრულებელ ოპერაციებს;
- განსაზღვრავს მენსიერების იმ უჯრედთა მისამართებს, რომლებშიც საწყისი მონაცემებია მოთავსებული;
- **ALU**-ში შეაქვს საწყისი მონაცემები;
- მართავს ოპერაციის შესრულების პროცესს;
- შეინახავს მიღებულ შედეგს.

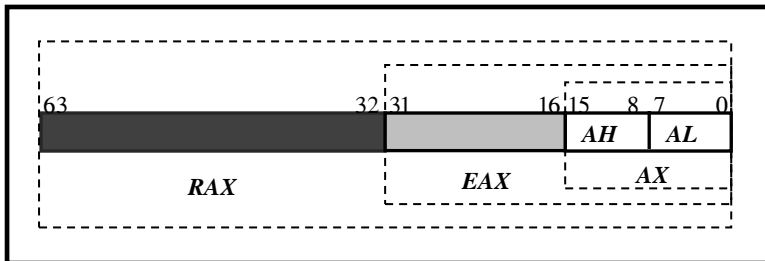
ზემოთ ჩამოთვლილი ელემენტალური ოპერაციებს შესასრულებლად განკუთვნილი თითოეული **სამანქანო ბრძანება** შედგება ელემენტარული მოქმედებების განმსაზღვრელი ბრძანებებისაგან, რომლებსაც **მიკრობრძანებები** ეწოდება.

სხვადასხვა სამანქანო ბრძანების შესასრულებლად შეიძლება სხვადასხვა რაოდენობის ელემენტალური ქმედების შესრულება იყოს საჭირო. რაც უფრო მეტია შესასრულებელი ელემენტალური ქმედებების რაოდენობა, მით უფრო მეტია მის შესასრულებლად საჭირო მიკრობრძანებების რაოდენობაც. მაგალითად, მიკროპროცესორის ერთი შიგა რეგისტრიდან მეორეში გადასაგზავნად გაცილებით მცირე რაოდენობის ქმედებებია შესასრულებელი, ვიდრე ორი მონაცემის ერთმანეთზე გადასამრავლებლად. ოპერატიულ მენსიერებასთან მომუშავე ბრძანებების შესასრულებლად უფრო მეტი დროა საჭირო, ვიდრე მხოლოდ პროცესორის რეგისტრებთან მომუშავე ბრძანებების შესასრულებლად.

სამანქანო ბრძანების თითოეული მიკრობრძანებას აამუშავებს მმართველი იმპულსი. მართვის მოწყობილობა ამ მიზნისათვის საჭირო იმპულსებს იღებს **ტაქტური იმპულსების გენერატორისაგან**. ორ მეზობელ იმპულსს შორის არსებულ ინტერვალს **ტაქტი ეწოდება**.

ორი მიკრობრძანება თუ სრულებით არ არის ერთმანეთზე დამოკიდებული, მაშინ ისინი შეიძლება ერთდროულად (ერთი ტაქტის განმავლობაში) შესრულდეს მაშინაც კი, როდესაც ისინი პროგრამის სხვადასხვა ბრძანებას მიეკუთვნება. თანამედროვე პროცესორებში სწრაფმოქმედების ასამაღლებლად ასეთი ოპტიმიზაცია ფართოდ გამოიყენება კონვეიერული დამუშავების ორგანიზებისათვის.

**3. პროცესორის რეგისტრები** არითმეტიკულლოგიკური და მართვის მოწყობილობების გარდა მიკროპროცესორში სხვა უამრავი რეგისტრებიცაა. მათი უმრავლესობა პროცესორის შიგნითაა მოთავსებული და მიუწვდომელია დამპროგრამებლისათვის. მათ გარდა არსებობს სპეციალურად პროგრამული უზრუნველყოფის მიერ გამოსაყენებლად განკუთვნილი **რამდენიმე რეგისტრიც**. შესასრულებელი ფუნქციების უნივერსალობისათვის ხაზგასასმელად ასეთ რეგისტრებს ხშირად **საერთო დანიშნულების რეგისტრებს (სღრ-ებს)** უწოდებენ. **სღრ-ებში** შეიძლება ინახებოდეს არა თავად მონაცემები (რიცხვები, სიმბოლოთა კოდები და ა. შ.), არამედ მეხსიერების იმ უჯრედების მისამართებიც, რომლებშიც ეს მონაცემებია მოთავსებული. მაგალითად, მეხსიერების თუ მიმღევრობითი უჯრედების დამუშავება მოითხოვება, მაშინ ასეთი რეგისტრის შიგთავსს ყოველთვის უჯრედის ზომა უნდა დაემატოს.



**ნახ. 2.5.** ჩალაგებული სტრუქტურის 64-ბიტიანი რეგისტრი

სხვადასხვა პროცესორში სხვადასხვა რაოდენობისა და განსხვავებული სტრუქტურის რეგისტრებია. მაგალითად **Intel**-ის ოჯახის

პროცესორებში მცირე რაოდენობის **64**-თანრივიანი რეგისტრია. ძველი (**32**- და **16**-თანრივიანი) პროცესორებთან პროგრამული შეთავსებადობის უზრუნველსაყოფად საერთო დანიშნულების ამ რეგისტრებს რუსული „მატრიოშკას“ მსგავსი ჩალაგებული სტრუქტურა აქვს (ნახ. **2.5**).

**2.5** ნახაზზე ნაჩვენებია **64**-თანრივიანი **RAX** რეგისტრი. მისი უმცროსი **32** ბიტი (ნულოვანი ბიტიდან **31**-ე ბიტამდე) წარმოქმნის **32**-თანრივიან **RAX** რეგისტრს. ამ უკანასკნელის უმცროსი **16** ბიტი (**0**-დან **15**-მდე) თავის მხრივ წარმოქმნის **16**-თანრივიან **EAX** რეგისტრს, და, ბოლოს, **0**-დან **7**-მდე და **8**-დან **15**-მდე ბიტები წარმოქმნის **8**-თანრივიან **AL** და **AH** რეგისტრებს. ნათლად ჩანს, რომ **Intel** პროცესორების თანრივიანობა თანდათანობით იზრდებოდა. რეგისტრების ასეთი სტრუქტურა უზრუნველყოფს წინა მოდელებთან თავსებადობას და პროცესორს საშუალებას აძლევს ადვილად დაამუშაოს **8**-, **16**-, **32**- და **64**-თანრივიანი მონაცემები.

ზემოთ განხილული **RAX** რეგისტრის გარდა **intel**-ის **64**-თანრივიან პროცესორებში არის ანალოგური სტრუქტურის **RBX**, **RCX** და **RDX**, აგრეთვე ზოგიერთი სხვა რეგისტრი. ეს რეგისტრები ერთმანეთის ტოლფასი არ არის და საჭიროა სპეციალური ცნობარით დავადგინოთ თითოეულ მათგანთან რომელ ბრძანებას შეუძლია იმუშაოს.

#### **4. კემქსსი მრეგება და თანაპროცესორი კემქსსიერება**

ოპერატიული მეხსიერების მიერ ხშირად გამოყენებადი მონაცემებისა და პროგრამის ნაწილის შესანახი ზესწრაფი მეხსიერებაა. პროცესორი საჭირო მონაცემებს თავდაპირველად **კემქსსიერებაში** მოიძიებს და ისინი თუ იქ ვერ აღმოაჩინა, შემდეგ გააგრძელებს ძებნას უფრო ნელმოქმედ ოპერატიულ მეხსიერებაში. **კემქსსიერება** უფრო ხშირად ორ, ხოლო იშვიათად – სამ დონედ იყოფა რომლებიც შესაბამისად აღინიშნება როგორც **L1**, **L2** და **L3**.

**ალმ**, როგორც წესი, ოპერაციებს მხოლოდ მთელ რიცხვებზე ასრულებს. ნამდვილ რიცხვებზე ოპერაციების ჩასატარებლად გამოიყენება მთემატიკური **თანაპროცესორი**, რომელიც თანამედროვე მიკროპროცესორის ბირთვშია ჩაშენებული (იხ. ნახ. **2.4**).

**მიკროპროცესორის  
მასსიათებლები.**

მიკროპროცესორის ძირითადი მასსიათებლებებია ტაქტური სიხშირე, თანრივიანობა, კემქსსიერების მოცულობა და ბირთვების

რაოდენობა. განვიხილოთ ისინი.

**1. ტაქტური სინშირე.** მიკროპროცესორს სხვადასხვა ბრძანების შესასრულებლად სხვადასხვა რაოდენობის ტაქტი სჭირდება, ე. ი. მათ შესრულებაზე სხვადასხვა დროს ხარჯავს. ტაქტურ იმპულსების გენერირებას, როგორც ზემოთ აღვნიშნეთ, ახდენს სისტემურ დაფაზე დაყენებული ტაქტური სინშირის გენერატორი. რაც უფრო ხშირად გენერირდება ისინი, პროცესორი დროის ერთეულში მით უფრო მეტ ბრძანებას ასრულებს, ე. ი. მით უფრო მეტია მისი სწრაფმოქმედება. ტაქტურ სინშირე ჩვეულებრივად მეგაჰერცებით გამოისახება. **1** მეგაჰერცის წამში **1** მილიონი ტაქტის ტოლია. **Intel**-ის მიერ დამუშავებული პროცესორების ადრეული **i8008x** მოდელები **5** მეგაჰერცზე ნაკლები ტაქტური სინშირით მუშაობდა. თანამედროვე პროცესორების ტაქტური სინშირე **3** გიგაჰერცს აღემატება (**1** გიგაჰერცი = **1000** მეგაჰერცს).

პროცესორის მუშაობაზე მისი **შიგა არქიტექტურა** ისეთივე გავლენას ახდენს, როგორც მისი ტაქტური სინშირე; ამიტომ სრულებით არაა სავალდებულო სხვადასხვა არქიტექტურისა და ერთნაირი ტაქტური სინშირის მქონე ორმა მიკროპროცესორმა ერთი და იგივე ბრძანება ერთნაირ დროში შესასრულოს. მაგალითად, **i80286** მიკროპროცესორს ორი რიცხვის გასამრავლებლად **20** ტაქტი სჭირდება, მაგრამ იმავე ოპერაციას **i80486** და მოდევნო თაობის პროცესორები ერთ ტაქტში ასრულებს. უფრო მეტიც, არსებობს **1** ტაქტში ერთზე მეტი ბრძანების შემსრულებელი მიკროპროცესორებიც.

ერთმანეთისაგან განასხვავებენ გარე და შიგა ტაქტურ სინშირეს.

**გარე ტაქტური სინშირე** ეწოდება სინშირეს, რომლითაც პროცესორი მონაცემებს უცვლის კომპიუტერის ოპერატიულ მეხსიერებას. მის გენერირებას ახდენს ტაქტური იმპულსების გენერატორი (კვარცული რეზონატორი).

**შიგა ტაქტური სინშირე** ის სინშირეა, რა სინშირითაცაა ორგანიზებული მუშაობა პროცესორის შიგნით.

ადრეულ პროცესორებს ერთნაირი გარე და შიგა ტაქტური სინშირე ჰქონდა, მაგრამ **i80486** პროცესორიდან შიგა ტაქტური სინშირის განსასაზღვრად დაიწყო **გამრავლების კოეფიციენტის**

გამოყენება. ეს კოეფიციენტი განისაზღვრება ცენტრალური პროცესორის გარკვეულ კონტაქტებზე ძაბვის მიწოდებით. ამგვარად, თანამედროვე მიკროპროცესორის შიგა ტაქტური სიხშირე მიიღება

ტექნოლოგიური ნორმები      პროცესორული ინტერფეისი  
 ბირთვის სახელი              პროცესორის სახელი

**CPU-Z**

CPU   Cache   Mainboard   Memory   SPD   About

Processor

Name: Intel Celeron M 560  
 Code Name: Conroe-L      Brand ID:   
 Package: Socket 479 mPGA  
 Technology: 65 nm      Core Voltage:

Specification: Intel(R) Celeron(R) CPU 560 @ 2.13GHz

Family	6	Model	6	Stepping	1
Ext. Family	6	Ext. Model	16	Revision	A1

Instructions: MMX, SSE, SSE2, SSE3, SSSE3, EM64T

Clocks (Core #0)

Core Speed	2128.1 MHz
Multiplier	x 16.0
Bus Speed	133.0 MHz
Rated FSB	532.0 MHz

Cache

L1 Data	32 KBytes
L1 Inst.	32 KBytes
Level 2	1024 KBytes
Level 3	

Selection: Processor #1      Cores: 1      Threads: 1

Version 1.49

**CPU-Z**      OK

გარე ტაქტური სიხშირე  
 გამრავლების კოეფიციენტი  
 შიგა ტაქტური სიხშირე

ბირთვების რაოდენობა  
 მე-2 დონის კეშის ზომა  
 1-ლი დონის კეშის ზომა

ნახ. 2.6. CPU-Z პროგრამის ინტერფეისი

ზემოთ აღნიშნულ კოეფიციენტზე გარე ტაქტურ სიხშირის გამრავლებით. მაგალითად, თუ გარე ტაქტური სიხშირე **133** მგჰც-ის, ხოლო გამრავლების კოეფიციენტი **10**-ის ტოლია, მაშინ შიგა ტაქტური სიხშირე იქნება  $133 \cdot 10 = 1330$  მგჰც.

**2. პროცესორის თანრიგიალება** განისაზღვრება მონაცემთა ბიტების რაოდენობით, რომლებიც მან შეიძლება ერთდროულად შეასრულოს. პირველი პროცესორები იყო **8-** და **16-**თანრიგიანები, ხოლო თანამედროვე პროცესორები **32-** და **64-**თანრიგიანებია.

**3. კაშხის სიჩქარის მოცულობა** როგორც ზემოთ აღვნიშნეთ, საჭირო ინფორმაციის ძიებას პროცესორი **კაშ-მეხსიერებიდან** იწყებს. რაც უფრო დიდია ამ უკანასკნელის მოცულობა, მით უფრო მაღალია იმის ალბათობა, რომ საჭირო ინფორმაციას პროცესორი სწორედ **კაშ-მეხსიერებაში** იპოვნის და მის ძიებაზე დიდ დროს არ დახარჯავს, რაც აამაღლებს მის სწრაფმოქმედებას.

**4. ტექნოლოგიური ნორმები.** ტექნოლოგიური ნორმები განსაზღვრავს მანძილს მეზობელ ტრანზისტორებს შორის. რაც უფრო მცირეა მანძილი, მით უფრო მოკლეა ტრანზისტორების არხები და მით უფრო მაღალია მათი სწრაფქმედება. გარდა ამისა, მანძილების შემცირება ამდაბლებს სითბოს გამოყოფის სიმძლავრეს. დღეს პროცესორების ასაგებად გამოიყენება **0,09**, **0,065** და **0,045** მიკრონიანი ტექნოლოგიური ნორმები (**1** მიკრონი =  $10^{-6}$  მეტრს). ზოგჯერ ტექნოლოგიურ ნორმებს ნანომეტრებშიც გამოსახავენ (**1 ნმ** =  $10^{-9}$  მ).

**5. ბირთვების რაოდენობა** თანამედროვე პროცესორების უმრავლესობა რამდენიმე (ჩვეულებრივად ორიდან ოთხამდე) ბირთვს შეიცავს. რამდენიმე ბირთვის არსებობის წყალობით პროცესორს შეუძლია ერთდროულად ბრძანებების რამდენიმე ნაკადი დაამუშაოს, ე.ი. რეალური დროის რეჟიმში რამდენიმე ამოცანა პარალელურად გადაწყვიტოს.

პროცესორის ძირითადი პარამეტრების განსასაზღვრავად შეიძლება სპეციალური სერვისული პროგრამა გამოვიყენოთ. ასეთი პროგრამაა, მაგალითად, **CPU-Z** (ნახ. **2.6**), რომელიც შეიძლება ინტერნეტიდან უფასოდ გადმოვწეროთ.



## 2.6. პირსონალური კომპიუტერის მისიერება

პროგრამის *ბრძანებები*, რომლებიც პროცესორმა უნდა შეასრულოს და *მონაცემები*, რომლებზეც მან უნდა მოახდინოს ოპერირება, *შენახება მეხსიერებაში*. აღნიშნულიდან გამომდინარე, მეხსიერების გარეშე ვერ ავაგებთ გამომთვლელ ავტომატს. ამასთანავე (რაც არანაკლებ მნიშვნელოვანია) იგი დასაშუშავებელი მონაცემების შესანახადაც გამოიყენება.

*მეხსიერება* წარმოადგენს მოწყობილობას, რომელშიც ჩაიწერება შეინახება და მოთხოვნის შესაბამისად გაიცემა პროგრამის ბრძანებები და მონაცემები.

არსებობს მეხსიერების მრავალი სახე, რომლებიც ერთმანეთისაგან სხვადასხვა ნიშნით განსხვავდება. მათი სიმრავლე იყოფა ორ დიდ ჯგუფად – *შივა* და *გარე მეხსიერება*. ეს ტერმინები ისტორიულად ჩამოყალიბდა იმის გამო, რომ ადრეულ კომპიუტერებში მეხსიერების ერთი ნაწილი განთავსებული იყო პროცესორის შესანახი კარადის შიგნით (*შივა მეხსიერება*), ხოლო მეორე ნაწილი კი – ამ კარადის გარეთ (*გარე მეხსიერება*).

თანამედროვე კომპიუტერები გარეგნულად იმდენად შეიცვალა, რომ დაკარგა საკუთარი ადრეული თვალსაჩინოება. მიუხედავად ამისა, მეხსიერების სიმრავლის ზემოთ მოყვანილ ჯგუფებად დაყოფა დღემდე იქნა შენარჩუნებული. ეს ჯგუფები ახლა ერთმანეთისაგან არა ადვილმდებარეობის, არამედ დანიშნულების მიხედვით განსხვავდება. *შივა მეხსიერება* ეწოდება მოცემულ მომენტში გადასაწყვეტი ამოცანებისათვის საჭირო პროგრამებისა და მონაცემების შესანახ მეხსიერებას, ხოლო *გარე მეხსიერება* – მეხსიერებას, რომელშიც ხანგრძლივად შეინახება მომავალში სავარაუდოდ წარმოსაშვები ამოცანების გადასაწყვეტად საჭირო პროგრამები და მონაცემები. აღნიშნული სპეციფიკურობის გამო გარე მეხსიერებას ხშირად *ხანგრძლივი შენახვის მეხსიერებასაც* უწოდებენ.

**შივა მეხსიერების  
ზოგადი დახასიათება**

შივა მეხსიერებას ხშირად *ძირითად მეხსიერებასაც* უწოდებენ. მის შემადგენლობაში შედის *ოპერატიული დამხსო* – მებელი მოწყობილობა (*ოლმ*) და *მუდმივი დამხსომებელი მოწყობილობა* (*მლმ*).

შიგა მეხსიერება აიგება **1.4** პარაგრაფში მოყვანილი საბაზისო პრინციპების შესაბამისად (გვ. **26-31**). გარე მოწყობილობისაგან ძირითადად იგი იმით განსხვავდება, რომ მისი ცალკეული უჯრედის მისამართების წყალობით თავისუფლად შეიძლება ნებისმიერ უჯრედთან შედგება.

**MLM**-ში შენახული ინფორმაცია ითვლება დროებით (ოპერატიული) მეხსიერებად, ამიტომ მომხმარებელმა მისთვის საჭირო მონაცემები თვითონვე უნდა შეინახოს გარე მეხსიერებაში.

ითვლება, რომ კვების ამორთვის შემდეგ იკარგება **MLM**-ში არსებული ინფორმაცია. უფრო ზუსტად თუ ვიმსჯელებთ, ეს მთლად სწორი არ არის, რადგან არსებობს მეხსიერების ელემენტები, რომლებსაც დენის ამორთვის შემდეგაც შეუძლია საკუთარი მდგომარეობის შენარჩუნება. საქმე ისაა, რომ კომპიუტერის განმეორებით ჩართვის (ან გადატვირთვის) დროს **პროგრამულ უზრუნველყოფას** არა აქვს უნარი აღადგინოს კვების ამორთვამდე სად რომელი ინფორმაცია ინახებოდა. ამიტომ ტექსტის აკრეფის დროს, კომპიუტერს თუ გადავტვირთავთ, მაშინ სამუშაოს ხელახლა შესრულება მოგვიხდება.

შიგა მეხსიერება მნიშვნელოვნად განსხვავებული ტექნოლოგიების გამოყენებით შეგვიძლია ავაგოთ. **პირველი თაობის კომპიუტერებში** გამოიყენებოდა **ელექტრონული მილაკებით აგებული ოპერატიული დამხსოვებელი მოწყობილობები**; ამასთანავე, მის ასაგებად გამოყენებული მილაკების რაოდენობა მეხსიერების თანრიგების რაოდენობას ემთხვეოდა (რიცხვის თითოეული ბიტი განცალკევებული მილაკიდან წაიკითხებოდა). შემდეგ გამოჩნდა **მაგნიტური გულარებით აგებული მეხსიერება**. გულარის დამაგნიტებულ მდგომარეობას შეესაბამებოდა ბიტის **1**-ის ტოლი მნიშვნელობა, ხოლო დაუმაგნიტებელს – **0**-ის მდგომარეობა. აღსანიშნავია ის, რომ მეხსიერების მაგნიტურ უჯრედებში მონაცემები კვების ამორთვის შემდგომაც სრულად შეინახებოდა. საბოლოოდ, მიკროელექტრონიკის განვითარებამ საშუალება მოგვცა დაგვემზადებინა **კომპაქტური ნახევარგამტარული მეხსიერება**, რომელიც თანამედროვე პერსონალურ კომპიუტერებშია გამოყენებული.

არსებობს **სტატიკური და დინამიკური მეხსიერება**, რომლებიც ერთმანეთისაგან განსხვავებული ტექნოლოგიებით აიგება. პირველი

მათგანი აიგება *ტრიგერებით*, ხოლო მეორე მათგანი – *ნახევარგამტარული კონდენსატორებით*. კონდენსატორი ტრიგერზე გაცილებით მარტივი და მცირე ზომისაა, რის გამოც ერთსა და იმავე კრისტალზე სტატიკურ მეხსიერების ელემენტებზე გაცილებით მეტი რაოდენობის დინამიკური მეხსიერების ელემენტები ეტევა. ამიტომ *დინამიკური მეხსიერების* ტევადობა აღემატება *სტატიკური მეხსიერების* ტევადობას და მასზე იაფიცაა. სამწუხაროდ, *დინამიკურ მეხსიერებას* აქვს მნიშვნელოვანი ნაკლი: იგი სტატიკურ მეხსიერებაზე გაცილებით ნელმოქმედია. დღეისათვის პერსონალურ კომპიუტერებში დინამიკური ოპერატიული მეხსიერება გამოიყენება. შევნიშნავთ, რომ კომპიუტერში დინამიკური მეხსიერება პირველად შექმნა ბულგარელი წარმოშობის ამერიკელმა მეცნიერმა *ჯონ ათანასოვმა*, რომელიც პირველი ელექტრონულ გამოთვლელი მანქანის შემქმნელადაც მიიჩნია ამერიკულმა სასამართლომ.

*მღმ*-ის დამზადების ტექნოლოგიის სრულყოფაც ასევე თანდათან ხდებოდა. მასში ინფორმაცია თავდაპირველად მხოლოდ ქარხანაში შეიტანებოდა. შემდეგ გამოჩნდა *დაპროგრამებადი მღმ*; მომხმარებელს შეეძლო მისი „სუფთა“ („ცარიელი“) მიკროსქემა მოეთასებინა სპეციალურ მოწყობილობაში – *პროგრამატორში* და თვითონ ჩაეტვირთა მასში ინფორმაცია. ამ ტიპის ზოგიერთ მიკროსქემებში დამხსომებელ ელემენტებად გამოიყენებოდა წვრილი დენგამტარი *ზღუდარები*. ზღუდარის არსებობა შეესაბამებოდა ლოგიკურ *I*-ს. პროგრამატორი დენის მძლავრი იმპულსებით გადაწვავდა საჭირო ზღუდარებს და ამით ბიტებს დააყენებდა ლოგიკური *0*-ის მდგომარეობაში. ნათელია, რომ ასეთი ხერხით ინფორმაცია მხოლოდ ერთჯერადად ჩაიწერებოდა და შემდეგ მისი შეცვლა შეუძლებელი იყო.

მოგვიანებით გამოჩნდა *გადაპროგრამებადი მღმ*-ები, რომლებშიც შეგვეძლო როგორც ჩაგვეწერა, ასევე წაგვეშალა ინფორმაცია. თავდაპირველად ჩაწერილი ინფორმაციის წაშლა შეიძლებოდა ულტრაიისფერი სხივების, ხოლო შემდეგ – ელექტრული იმპულსების დახმარებით. თანამედროვე გადაპროგრამებადი დამხსომებელი მოწყობილობა იყენებს ფლემ-მეხსიერებას. ასეთი მეხსიერების თითოეული ელემენტი განსაკუთრებული ნაირსახეობის ტრანზისტორებითაა აგებული, ამიტომ იგიც ნახევარგამტარულ მეხსიერებაა. მისი

შიგთავსის შეცვლა პროგრამატორის გარეშე – სპეციალური პროგრამითაც შეგვიძლია.

კომპიუტერი, როგორც წესი, შეიცავს **BIOS**-ის მიკროსქემას, რომელშიც ჩაწერილია ჩაშენებული პროგრამული უზრუნველყოფა – პროგრამების ნაკრები, რომელიც, კომპიუტერის ჩართვის შემდეგ შეამოწმებს აპარატურას, კომპიუტერში ჩატვირთავს საწყის ინფორმაციას და გარკვეულ მოწყობილობებთან (მაგალითად, კლავიატურასთან, მონიტორთან, დისკებთან) გაცვლის საჭირო ინფორმაციას. კომპიუტერთა **IBM PC** ოჯახში ასეთ ჩაშენებულ პროგრამულ უზრუნველყოფას ეწოდება **BIOS** (**B**asic **I**nput/**O**utput **S**ystem – შეტანა/გამოტანის საბაზისო სისტემა).

**IBM PC**-შეთავსებად კომპიუტერებში კიდევ ერთი განსაკუთრებული სახის მეხსიერება გამოიყენება, რომელსაც **კონფიგურაციის მეხსიერება (CMOS-მეხსიერება)** ეწოდება (**C**omplementary **M**etal-**O**xide-**S**emiconductor – ლითონ-ოქსიდური-ნახევარგამტარის კომპლემენტარული სტრუქტურა). მასში შენახულია აპარატურული უზრუნველყოფის მრავალფეროვანი მომმართველები, აგრეთვე საათი და კალენდარი, რომელთა წყალობითაც კომპიუტერმა ყოველთვის «იცის» მიმდინარე თარიღი და დრო. მასში მონაცემები მცირე ბატარეიდან მიწოდებული კვების მეშვეობით შეინახება. **CMOS-მეხსიერება** განსაკუთრებული მეხსიერებაა, რომელიც არ შედის შიგა მეხსიერების სამისამართო სივრცეში. ამიტომ იგი უფრო გარე მეხსიერებას ჩამოჰკავს. კონფიგურაციის მეხსიერებასთან მუშაობისათვის თანამედროვე პერსონალური კომპიუტერის **BIOS**-ში **BIOS Setup** სახელწოდების სპეციალური პროგრამაა გათვალისწინებული, რომელთანაც მომხმარებელს შეუძლია ოპერაციული სისტემის ჩატვირთვამდე (კომპიუტერის ჩართვის დროს) იმუშაოს.

### ოპერატიული მეხსიერება\*

ოპერატიული მეხსიერება პერსონალური კომპიუტერის ერთ-ერთი ძირითადი ნაწილია რომელიც მნიშვნელოვან გავლენას ახდენს მის მწარმოებლურობაზე და ამიტომ იგი რამდენადმე დაწვრილებით განვიხილოთ.

ოპერატიული დამხსომებელი მოწყობილობა მიკროპროცესორისა და სხვა მოწყობილობებისათვის საჭირო ინფორმაციის შესანახად განკუთვნილი ძირითადი ნახევარგამტარული მეხსიერებაა. იგი კომ-

პიუტერის მუშა მესხიერებაა, რომლის სიდიდე განსაზღვრავს ერთ-დროულად შესრულებადი პროგრამების რაოდენობასა და ზომას. ოპერაციული სისტემის ან გამოყენებითი პროგრამების ამუშავებისას მონაცემები გარე მესხიერებიდან ჩაიტვირთება ოპერატიულ მესხიერებაში. ოპერატიული მესხიერებისათვის დამახასიათებელი უმნიშვნელოვანესი თვისებაა **ენერგოდამოკიდებულება**: კომპიუტერის კვების ამორთვისას მასში არსებული ყველა მონაცემი იკარგება.

წარმოება ოპერატიულ მესხიერებას უშვებს **მესხიერების მოდულებს** (*Memory module*) სახით. ყველაზე მეტად გავრცელებულია მესხიერების შემდეგი ორი სახის მოდული:

■ **მესხიერების მოდული SIMM**, რომელშიც კონტაქტები მხოლოდ ერთ მხარეზეა განლაგებული (**Singl In-line Memory Module** – მესხიერების ცალმხრივი მოდული). იგი ფართოდ გამოიყენებოდა **1990**-იან წლების კომპიუტერებში;

■ **მესხიერების მოდული DIMM**, რომელშიც კონტაქტები ორ მოპირდაპირე მხარეზეა განლაგებული (**Dual In-line Memory Module** მესხიერების ორმხრივი მოდული); მის გამოჩენას ხელი შეუწყო პროცესორ **Pentium**-ის გამოყენებამ, რომელსაც ჰქონდა მონაცემების **64**-თანრივანი სალტე. მთელ რივ ნოვაციებთან ერთად მოცემულ მოდულს აქვს შედომების გამომჟღავნებისა და გასწორების უნარი.

პერსონალური კომპიუტერის ოპერატიული მესხიერების მთელი მოცულობა დაყოფილია **მესხიერების რამდენიმე ბანკად** (*Memory bank*), რომლებშიც გამოყენებული მოდულების სახე და ტიპი **სისტემურ (დედა) დაფაზე** დამოკიდებული (შესაბამისი სპეციფიკაციები საექსპლუატაციო ცნობარშია მოყვანილი). ბანკის თანრივიანობა შეესაბამება დაყენებული მიკროპროცესორის მონაცემების სალტის თანრივიანობას, ხოლო მესხიერების თითოეული მოდულის დასამისამართებლად სხვადასხვა რაოდენობის ბიტებია გამოყენებული. კერძოდ, **30**-კონტაქტური **SIMM** მოდულის დასამისამართებლად გამოყენებულია **8** ბიტი, **72**-კონტაქტური **SIMM** მოდულის დასამისამართებლად – **32** ბიტი, ხოლო **168** კონტაქტური **DIMM** მოდულის დასამისამართებლად – **64** ბიტი.

ოპერატიული დამხსომებელი მოწყობილობის უმნიშვნელოვანესი პარამეტრია **შელწვეის დრო**, რომელიც აუცილებელია იმ ინფორმაციასთან მიმართვის სრული ციკლის განსახორციელებლად, რომელიც მესხიერების შემთხვევით ადგილზეა შენახული. პირველ პერსონალურ კომპიუტერებში ოპერატიული მესხიერებასთან შელწვეის დრო თუ **140** ნანოწამზე მეტი იყო,

თანამედროვე პერსონალური კომპიუტერების ნორმალური მუშაობისათვის საჭიროა ეს დრო **5** ნანოწამის ფარგლებში იყოს. მეხსიერებასთან შეღწევის დრო და მიკროსქემების რაოდენობა ჩვეულებრივ მოდულის კორპუსზეა მითითებული.

არსებობს შემდეგი შვიდი ტიპის ოპერატიული მეხსიერება:

**1. FRM**-ტიპის მეხსიერება (**F**ast **P**age **M**ode – სწრაფი გვერდობრივი რეჟიმი) საშუალებას გვაძლევს მნიშვნელოვნად ვაკუზარდოთ მეხსიერების მიმდევრობითად განლაგებულ უჯრედებთან შეღწევის დრო. დინამიკური მეხსიერების მატრიცაში სტატიკური ბუფერი მეხსიერების მთელ სტრიქონს ერთბაშად წაიკითხავს, ხოლო საჭირო ბიტი უკვე სვეტის მისამართზე დამოკიდებულებით ამოირჩევა; ამიტომ მორიგი წასაკითხი ბიტი უკვე თუ წაკითხულ სტრიქონშია, მაშინ ბუფერში მისი კიდევ ერთხელ გადატანა საჭირო არ არის. ამ ტიპის მეხსიერება მოთავსებული იყო **SIMM** მოდულეში და გამოიყენებოდა ძველ პერსონალურ კომპიუტერებში (შელწევის დრო **70** ნანოწამი);

**2. EDO**-ტიპის მეხსიერება (**E**xtended **D**ata **O**ut – გაფართოებული მონაცემები) უკანასკნელად მოთხოვნილ მონაცემებს გამოყენების შემდეგ ინახავს **კეშ-მეხსიერებაში**, რომლის არსს ქვემოთ ვავეცნობით. ეს **20%**-ით ზრდის ოპერატიული დამხსოვებელი მოწოდებლობის მწარმოებლურობას. ამ ტიპის მეხსიერება მოთავსებული იყო **SIMM** მოდულებში და **1995** წლიდან გამოიყენებოდა კომპიუტერებში, რომლებშიც დაყენებული იყო **Pentium**-ის ტიპის მიკროპროცესორები. მოცემული ტიპის მეხსიერების დროს სალტის ტაქტური სიხშირე **83** მეგაჰერცამდე იყო;

**3. BEDO**-ტიპის მეხსიერებას (**B**urst. **E**DO – ფეთქებად გაფართოებული მონაცემები) აქვს სიტყვების სპეციალური მთვლელები. იგი ფართოდ ვერ გავრცელდა დაბალი (**66** მჰჰც-მდე) ტაქტური სიხშირის გამო;

**4. EDRAM** ტიპის მეხსიერება (**E**nhanced **D**RAM – გაფართოებული დინამიკური მეხსიერება) შეიცავს ჩვეულებრივი დინამიკური (**DRAM**) მეხსიერების უჯრედებსა და სწრაფმოქმედი (**SRAM**) სტატიკური მეხსიერების მცირე რაოდენობის უჯრედებს. ასეთი მეხსიერებაც ფართოდ ვერ გავრცელდა;

**5. SDRAM** ტიპის მეხსიერება (**S**ynchronous **D**RAM – სინქრონული დინამიკური მეხსიერება) პირველად იქნა გამოყენებული **Pentium**-ის უფროს მოდელებსა და **Pentium MMX**-ში. მოთავსებული იყო **DIMM** მოდულებში, რომელთა ტაქტური სიხშირე **66** მეგაჰერცადან იწყებოდა, შეღწევის დრო კი **60** ნანოწამზე არანაკლები იყო;

**6. DRDRAM** ტიპის მეხსიერება (**Direct Rambus DRAM** - პირდაპირი დინამიკური მეხსიერება) პირველად **2001** წელს გამოჩნდა **Pentium 4** მიკროპროცესორისათვის. იგი მოთავსებული იყო **400მპკ** ტაქტური სიხშირის მქონე **RIMM** მოდულში (პიკური გამტარობის უნარი **1,6** გიგაბაიტი წამში). მას მხარს უჭერდა კომპანია **Intel**, მაგრამ მაღალი ფასის გამო ფართოდ ვერ გავრცელდა;

**7. DDR SDRAM** ტიპის მეხსიერება (**Duble Data Rate SDRAM** - მონაცემების გაორმაგებული სიჩქარით გადაცემა სინქრონული მეხსიერება) დამუშავდა **DRDRAM** ტიპის მეხსიერების საპასუხოდ. იგი **SDRAM** მეხსიერების ახალ თაობას წარმოადგენს, რომელშიც შესაძლებელია მონაცემები ტაქტური სიხშირის სივრცის ორივე ფრონტიდან გადაიცეს, რაც აორმაგებს პიკური გადაცემის უნარს.

პროცესორების მწარმოებლობის მუდმივად ზრდა ოპერატიული მეხსიერების ზრდითაც უნდა იყოს მხარდაჭერული, რაც ძალიან რთულია. პროცესორების წარმოებაში მომხდარი რევოლუციურ ცვლილებათა ფონზე ოპერატიული მეხსიერების ევოლუცია ძალიან მოკრძალებულად გამოიყურება: შეიცვალა მხოლოდ ერთ კრისტალზე განთავსებული უჯრედების რაოდენობა, რამაც შეამცირა მონაცემების შენახვის კუთრი ღირებულება. ოპერატიული მეხსიერების საფუძველი კი პირველი პერსონალური კომპიუტერების გამოჩენიდან დღემდე არ შეცვლილა და კვლავ გამოიყენება კონდენსატორული დამხსომებელი უჯრედებიანი დინამიკური მეხსიერება, რომელიც **ათანასოვის** მიერ ჯერ კიდევ კომპიუტერული ტექნიკის განვითარების ვარიანტებზე იყო შემოთავაზებული.

**1990**-იანი წლების დასაწყისში გამოდიოდა კონკრეტული ოპერაციის დროის ფიქსირებულ პერიოდში შემსრულებელი **მეხსიერების ასინქრონული მოდულები**, რომლებიც კონკრეტულ ოპერაციას დროის ფიქსირებული შუალედში ამთავრებდა. მაშინ სპეციალისტებმა შეძლეს დაემუშავებინათ დაახლოებით **5მპკ** სიხშირეზე მომუშავე ოპერატიული მეხსიერება.

სტრიქონოვანი დამისამართების რეჟიმის დანერგვით შესაძლებელი გახდა ოპერატიულ **FRM** მეხსიერებაში ტაქტური სიხშირე **40 მპკ**-მდე გაზრდილიყო. მის შემდეგ დამუშავებული მის შემდეგ **EDO** მეხსიერება კომპიუტერული სისტემის მწარმოებლობის ამაღლების მიმართულებით გადადგმული ძორივი ნაბიჯი იყო. იგი ჩვეულებრივ **66მპკ** სიხშირეზე მუშაობდა და ჰქონდა **60** ნწ-მზე არანაკლები **მიმართვის ციკლი (მიმართვის ციკლი** წარმოადგენს მინიმალურ პერიოდს, რომლის განმავლობაშიაც ნებისმიერ მიმართზე მიმართვა შესაძლებელი). სამწუხაროდ, **EDO** მეხსიერებას **83 მპკ**-ზე უფრო მაღალ სიხშირეზე მუშაობა არ შეეძლო, ამიტომ საჭირო გახდა დამუშავებულიყო ახალი სახის მეხსიერება და ასეთ მეხსიერებად

მოკვევლინა **SDRAM** მეხსიერება, რომელიც თავდაპირველად მუშაობდა **66მგპც** სიხშირეზე. პირველმა ასეთმა მოდულებმა მრავალი პრობლემა შეუქმნა ავტოკონფიგურაციას (სისტემის მიერ მეხსიერების მოდულის ამოცნობას). მათი გადაწყვეტისათვის მეხსიერების **DIMM** მოდულზე განათავსეს მცირე **SPD**-ჩიბი (**S**erial **P**resence **D**etect), რომელიც მეხსიერების მოდულის ამოსაცნობად საჭირო ყველა ინფორმაციას გაცემდა.

ოპერატიული მეხსიერების ტექნოლოგიის განვითარების შედეგში ეტაპი დაკავშირებული იყო **1998** წელს გამოჩენილ და **2002** წლის დასაწყისამდე გამოყენებულ **Intel 440BX** ჩიპსეტთან კორექტულად მომუშავე მეხსიერების მოდულებისათვის კორპორაცია **Intel**-ის მიერ **PC-100** სპეციფიკაციის დაქვემდებარებასთან. ახალი მეხსიერების სწრაფმოქმედების აღწერისათვის **შელწვევის ციკლის ხანგრძლივობის** ნაცვლად შემოღებული იქნა **ტაქტური სიხშირის მინიმალური დასაშვები პერიოდი**, ე. ი. **10** ნანოწამით მოდულის მარკირებისას ივულისხმებოდა, რომ მას შეეძლო ემუშავა **100 მპც** სიხშირეზე; ზოლო ასეთი მეხსიერებისათვის **შელწვევის დრო** (დრო, რომელიც გადის მეხსიერებასთან მიმართვის მომენტიდან სისტემურ სალტზე მონაცემების გამოჩენამდე) დაახლოებით **50 ნწ**-მს შეადგენდა.

ოპერატიული მეხსიერების ტექნოლოგიის შემდგომ განვითარებაზე მნიშვნელოვანი გავლენა მოახდინა კორპორაცია **Intel**-მა, რომელსაც შემდეგი სამი პერსპექტიული **SDRAM PC-133** (იგი ტრადიციული **SDRAM** მწკრივის გამგრძელებელი იყო), **DRDRAM** და **DDRSDRAM** მეხსიერებიდან სტანდარტულ მეხსიერებად ერთ-ერთი მათგანი ამოერჩია და მისთვის დაეჭირა მხარი. სამწუხაროდ, მან წინდაუხედავად ამოირჩია **DRDRAM** მეხსიერება, რომელიც, როგორც ზემოთ აღვნიშნეთ, მაღალი ფასისა და სხვა დეტალების გამო ფართოდ ვერ გავრცელდა.

**1999** წელს მიელმა რიგმა მსხვილმა კომპანიამ (**IBM, Micron, NEC, Samsung** და **VIA**) მხარი დაუჭირეს ოპერატიული მეხსიერების **SDRAM PC-133** მეხსიერებას, რომლის მაქსიმალურმა გამტარობის უნარმა შეადგინა **1064 მმ/წმ**. მაგრამ პრაქტიკამ ვკინვენა, რომ ყველაზე პერსპექტიული ყოფილა **DDR SDRAM** მეხსიერება. მისი ღირსებაა ის, რომ მწარმოებლებისაგან მოითხოვს მცირე კაპიტალდაბანდებებს და, მამასადამე, საკმაოდ იაფია; ნაკლია ის, რომ **SDRAM** მოდულის გასართი (**168** კონტაქტი) არ ემთხვევა **DDR SDRAM** მოდულის გასართს (**182** კონტაქტი).

შემდეგ ელექტრონული მოწყობილობების დამამუშავებელი ორგანიზაციების **JEDEC** გაერთიანებამ (**J**oint **E**lectronic **D**evice **E**ngineering **C**ouncil) ინდუსტრიალ სტანდარტად ოფიციალურად დაამტკიცა **DDR** მეხსიერების **მეორე DDR2 თაობა**. მოდულებისათვის გამოიყენებოდა **240** კონტაქტის მქონე **BGA (FBGA)** ტიპის კორპუსი. სტანდარტის ჩარჩოებში დამუშავდა



შემდეგი სამი ტიპის მოდული: **PC2-4300 (533 მგ/ც სიხშირის ჩიპების დროს)**, **PC2-5300 (667 მგ/ც)** და **PC2-6400 (800 მგ/ც)**. გარდა ამისა, არსებობდა **1066 მგ/ც**-ზე მომუშავე სასინჯი ვერსიებიც.

**თანამედროვე კომპიუტერებში** გამოიყენება **მესამე თაობის ოპერატიული DDR3 მეხსიერების** მოდულები. მათაც **240** კონტაქტი აქვს, მაგრამ ისინი **DDR2**-თან შეუთავსებადია ელექტრულადაც და მექანიკურადაც. ამ სტანდარტის ჩარჩოებში დამუშავებულია და გამოდის შემდეგი **3** სახის მოდული: **PC3-8500 (1066 მგ/ც სიხშირის ჩიპების დროს)**, **PC3-10660 (1333 მგ/ც)** და **PC3-12800 (1600 მგ/ც)**. გარდა ამისა, არსებობდა **1866 მგ/ც**-ზე მომუშავე სასინჯი ვერსიებიც.

**DDR** მეხსიერების პირველი საცდელი ნიმუშები გამოჩნდა **1997** წელს, **DDR2** მეხსიერების საცდელი ნიმუშები – **2001** წელს, ხოლო **DDR3** მეხსიერების საცდელი ნიმუშები – **2005** წელს.

ალტერნატიული ტექნოლოგიების დამუშავებაზე მრავალი კომპანია მუშაობს. **2001** წელს **Fujitsu** კომპანიამ მონაცემების ხანგრძლივად შესანახად მასობრივად დაამზადა **სევენტოლექტრული მეხსიერება FRAM (Ferroelectric RAM)**. იგეგმება, რომ იგი **BIOS**-ში **EEPROM**-ის ნაცვლად (**E**lectrically **E**rasable **P**rogrammable **R**ead **O**nly Memory – ელექტრონული წაშლადი, დაპროგრამებადი, მარტო წასაკითხავი მეხსიერება) იქნეს გამოყენებული, რომელთან შედარებითაც მას აქვს ჩაწერის **10000**-ჯერ უფრო დიდი სიჩქარე და მოიხმარს **100**-ჯერ ნაკლებ ენერჯიას.

კომპანია **Intel** გეგმავს მომავალში ამორფული ნახევარგამტარული მეხსიერებისა (**Q**uonics **U**nified **M**emori - **OUM**) და პოლიმერული სევენტოლექტრული მეხსიერების (**P**olymeric **F**erroelectric **R**AM - **PFRAM**) ტექნოლოგიათა ბაზაზე დაამზადოს ენერგოდამოუკიდებელი მეხსიერება.

### გარე მეხსიერება

გარე მეხსიერება ეწოდება კომპიუტერის მეხსიერების ნაწილს, რომელშიც ხანგრძლივად შეინახება პროგრამები და მონაცემები.

ამ სახის მეხსიერება პროგრამისა და მონაცემების განმეორებით გამოყენების საშუალებას გვაძლევს. ამის წყალობით შეგვიძლია ერთხელ აკრეფილი ტექსტის ციფრული ფოტოსურათი მრავალი წლის განმავლობაში განვიხილოთ.

გარე მეხსიერების მოწყობილობებს ხშირად დამგროვებლებს უწოდებენ. არსებობს, მაგნიტური და ოპტიკური დისკებით ფორმირებული დამგროვებლები, აგრეთვე ნახევარგამტარული ფლემშემხსიერ-

ების საფუძველზე აგებული თანამედროვე გარე დამხსომებელი მოწყობილობები.

ნებისმიერი ტიპის გარე მეხსიერება შედგება ინფორმაციის გარკვეული **მზიდისაგან** (მაგალითად, დისკისაგან ან ნახევარგამტარული კრისტალისაგან) და მართვის ელექტრონული სქემისაგან (**კონტროლერისაგან**).

**ინფორმაციის კომპიუტერული მზიდი** კომპიუტერულ ფორმატში მონაცემების ხანგრძლივად შენახვის საშუალებაა. იგი შეიძლება იყოს მოსასხნელი (ასეთია ოპტიკური დისკები) ან მოთავსებული იყოს დაუშლადი მოწყობილობის შიგნით (ხისტი მაგნიტური დისკი).

კომპიუტერზე მუშაობის მოხერხებულობას მნიშვნელოვნად ამაღლებს **ხისტი მაგნიტური დისკები - HDD (Hard Disk Drive)**. ხშირად ხისტი დისკს **ვინჩესტერსაც** უწოდებენ. არსებობს ამ სახელწოდების წარმოშობის შემდეგი ვერსია. **IBM** ფირმის კონსტრუქტორთა ჯგუფმა **კენეტ ჰოტონის (Kenneth E. Haughton)** ხელმძღვანელობით **1973** წელს დაამუშავა **3340** მოდელის ხისტი დისკი. იგი შედგებოდა დაუშლად კორპუსში მოთავსებული დისკებისა და წამკითხავ თავისაგან. დამუშავების პროცესში კონსტრუქტორები მას უბრალოდ „**30-30**“ სახელით მოიხსენიებდნენ, რადგან შეიცავდა **30** მეგაბაიტთან ორი მოდულს. ეს სახელი შემთხვევით დაემთხვა პოპულარული სამონადირო მამხანა **ვინჩესტერში** გამოიყენებული ვაზნის სახელწოდებას (ამ ვაზნას უწოდებდნენ **30-30**), რის გამოც დამგროვებელმა ვინჩესტერის სახელწოდება მიიღო.

ხისტი დისკი წარმოადგენს სპეციალური მაგნიტომგონობიარე ნივთიერებით დაფარულ ლითონის ერთ ან ორ ფირფიტას, რომელიც წნეხილა ალუმინის ჰერმეტიკულ კორპუსშია მოთავსებული. გარდა ამისა, ვინჩესტერი შეიცავს ძრავას, წამკითხავ/ჩამწერ თავსა და მმართველ ელექტრონიკას. ხისტი დისკი მთითხოვს ძალიან ფრთხილ მოპყრობას, რადგან უბრალო რყევამ შეიძლება დააზიანოს მისი თავი.

პერსონალურ კომპიუტერში ხისტი დისკები პირველად მხოლოდ **1983** წელს გამოჩნდა. მაშინ მისი მოცულობა იყო **10** მეგაბაიტი, შელწვევის დრო – **100** მიკროწამი, ხოლო მონაცემების გადაცემის სიჩქარე – წამში **85** კილობაიტი. თანამედროვე ვინჩესტერის მოცულობა **160** გიგაბაიტიდან იწყება, მას აქვს **8-მეგაბაიტიანი** ბუფერი, **9** მიკროწამზე ნაკლები შელწვევის დრო და წამში გადასცემს **300** მეგაბაიტზე მეტ ინფორმაციას.

ხისტი დისკის ძირითადი პარამეტრებია ტვეალობა, საიმედოობა და მწარმოებლურობა. **საიმედოობა** ფასდება სტარტ/სდექ ციკლების რაოდენ-

ნობითა და მტყუნებაზე საშუალო ნამუშევრობით; გვაქვს ათასობით ციკლი, ხოლო საშუალო ნამუშევრობა ასობით ათასი საათით იზომება. **მწარმოებლობა** ფასდება წამში შესრულებული შეტანა/გამოტანის ოპერაციათა რაოდენობით, მონაცემების გადაცემის სიჩქარითა და პროცესორის ჩატვირთვით. პროცესორის ჩატვირთვა გვიჩვენებს, თუ რამდენ ტაქტს ხარჯავს პროცესორი ხისტ დისკზე მიმართვისათვის. რაც უფრო მცირეა იგი, მით უფრო ნაკლებ პრობლემებს უქმნის ხისტი დისკი სისტემას.

მონაცემების ნებისმიერ ბლოკთან შეღწევის უზრუნველსაყოფად მაგნიტური და ოპტიკური დისკები სწრაფად ბრუნავს, ხოლო წამკითხავი თავი დისკის რადიუსის გასწვრივ გადაადგილდება გარე მენსიერების უფრო თანამედროვე სახეებში, რომლებშიც ინფორმაციის მზიდად ნახევარგამტარული კრისტალთა გამოყენებული, არავითარი მოძრაობა ნაწილი არ არსებობს, ხოლო მონაცემების წაკითხვა-ჩასაწერად მხოლოდ ელექტრონული იმპულსები გამოიყენება (ოპერატიული დამხსომებელი მოწყობილობების ანალოგიურად).

გარე მენსიერების გადასატან მოწყობილობებში, მაგალითად, გარე ხისტ დისკებსა და ფლემდამგროვებლებში, მზიდი და მართვის სქემა ერთ ბლოკადაა გაერთიანებული. ასეთი მოწყობილობები კომპიუტერს გასართით გარედან უერთდება..

ცენტრალურ პროცესორს არ შეუძლია მზილზე არსებულ მონაცემებს უშუალოდ მიმართოს: ამას იგი მენსიერების კონტროლერით ახერხებს. **2.7** ნახაზზე ნაჩვენებია, თუ როგორ წაკითხება მონაცემები ინფორმაციის გარე მზიდიდან და ჩაიწერება ოპერატიულ დამხსომებელ მოწყობილობაში (**ოლგ**-ში). რეალურად გაცვლის პროცესი უფრო რთულია და მასში მენსიერებაში პირდაპირი დაშვების კონტროლერიც მონაწილეობს.

კონტროლერთან დასაკავშირებლად პროცესორი იყენებს **პორტებს** – კონტროლერის რეგისტრებს, რომლებსაც პროცესორს შეუძლია ნომრის (მისამართის) საშუალებით მიმართოს. პროცესორი კონტროლერს აძლევს მონაცემების წაკითხვის „დავალებას“, რომლის შემდეგ კონტროლერი თავის თავზე იღებს პროცესის მართვას. აღნიშნული პროცესის დამთავრებამდე პროცესორს პარალელურად შეუძლია განაგრძოს მუშაობა ან გადაწყვიტოს სხვა ამოცანა. ამგვარად, გარე მენსიერებიდან მონაცემების წაკითხვა (ავრეთვე ჩაწერა)

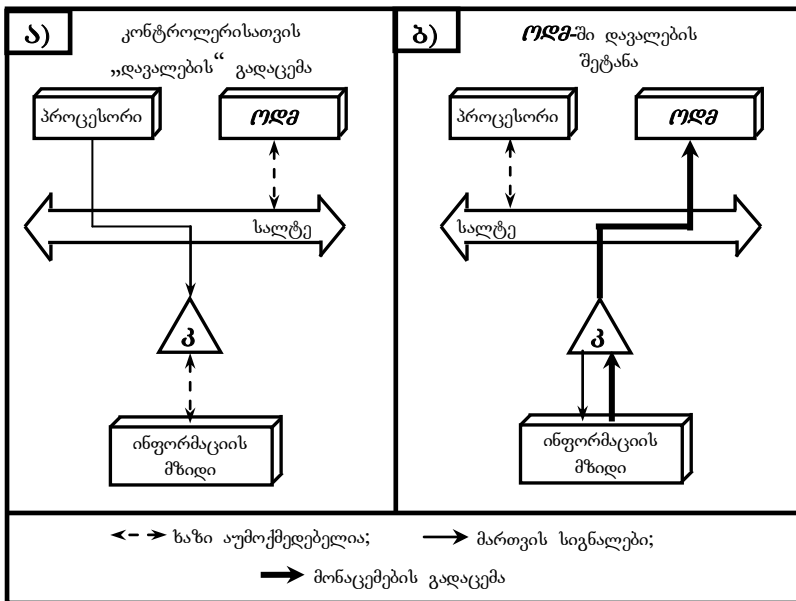
გაცილებით რთულია, ვიდრე შიგა მეხსიერებიდან იმავე ოპერაციების შესრულება.

გარე მეხსიერებისათვის დამახასიათებელია შემდეგი თავისებურებები:

- მონაცემები განთავსდება ბლოკებად (დისკებზე მათ *სექტორებს* უწოდებენ); მონაცემების ბლოკი წაიკითხება ან ჩაიწერება ერთბაშად, ე. ი. იგი ერთი მთლიანობის სახით განიხილება, რაც მნიშვნელოვნად აჩქარებს ინფორმაციის გაცვლის პროცესს. შეუძლებელია ბლოკის ნაწილთან მუშაობა;

- იმისათვის, რომ პროცესორმა შეძლოს უშუალოდ გამოიყენოს გარე მეხსიერებაში შენახული პროგრამა ან მონაცემები, საჭიროა ეს უკანასკნელები წინასწარ ჩაიტვირთოს ოპერატიულ მეხსიერებაში;

- მონაცემების გაცვლას მართავს კონტროლერები.



**ნახ. 2.7.** გარე მზიდიდან მონაცემების წაკითხვა და ოპერატიულ დამხსნობებელ მოწყობილობაში (ოღმ-ში) მისი ჩაწერა

გარე მეხსიერებად გამოიყენება სრულიად სხვადასხვა მზიდი. თავდაპირველად პროგრამები და მონაცემები შეინახებოდა ქაღალდის **პერფორაციებისა** და **პერფოლენტებზე**, რომლებსაც დამპროგრამებლები ხელით ახარისხებდნენ. შემდეგ ისინი შეცვალა ისეთმა **მაგნიტურმა მზიდებმა**, როგორებიცაა მაგნიტური ლენტები, დოლები და დისკები.

**მაგნიტურ დისკებზე** მონაცემთა ბიტები მცირე დამაგნიტებელი (ან განმაგნიტებელი) არეების სახით შეინახება. **სექტორები** განთავსდება საერთო ცენტრის მქონე კონცენტრირებულ წრეწირებზე, რომლებსაც **ბილიკები** ეწოდება. ადგილმდებარეობებზე დამოკიდებულით სხვადასხვა ბილიკის სიგრძე ერმანეთისაგან განსხვავდება, რის გამოც მათზე სხვადასხვა რაოდენობის სექტორები შეიძლება განთავსდეს. დისკის სექტორებში შესვლა თავისუფალია, მაქსიმალური სიჩქარე მაშინ მიიღწევა, როდესაც წასაკითხი და ჩასაწერი სექტორები ერთმანეთის მიყოლებითაა განთავსებული.

ძალიან შრომატევადია ასეთი რთული სისტემის მართვა, ამიტომ მაგნიტური დისკების გამოჩენის შემდეგ აუცილებელი გახდა დამუშავებულიყო სპეციალურად მათთან მუშაობისათვის საჭირო პროგრამული უზრუნველყოფა, რომელსაც **ოპერაციული სისტემა** ეწოდა. **ოპერაციული სისტემა** მომხმარებლებს საშუაოდ მონაცემების გარკვეულ ნაკრებებს – **ფაილებს** აწვდის, ხოლო ყველა დანარჩენი ტექნიკური დეტალების შესრულებას თავის თავზე იღებს: მომხმარებლები ფაქტობრივად მუშაობენ არა გარე მეხსიერებასთან, არამედ ფაილებთან. დისკური მზიდების შემოღების შემდეგ **ფაილური სისტემა** გახდა ის ძირითადი ნიშანი, რომლითაც გარე მეხსიერება განსხვავდება შიგა მეხსიერებისაგან.

მონაცემების შენახვის შემდგომ ტექნოლოგიას წარმოადგენს **ოპტიკური კომპაქტდისკები, CD (Compact Disk)**. მასზე მონაცემების ჩაწერის ერთ-ერთი ხერხის დროს ლაზერის სხივი დისკის ზედაპირზე „ამოწვავს“ ბილიკს, რომელზეც ურთიერთმონაცვლეობს ღრმულები და ბორცვები. ინფორმაციის წასაკითხავადაც ლაზერის სხივი გამოიყენება, ოღონდ იმისათვის, რომ მონაცემები არ დაირღვეს, იგი მცირე ინტენსივობის უნდა იყოს. ლოგიკური **0**-ებისა და **1**-ების ამოსაცნობად გამოიყენება დისკის ჩაღრმავებული და სწორი ზედაპირიდან ლაზერის სხივის განსხვავებულად არეკვლის ფაქტი.

მაგნიტური დისკებისაგან განსხვავებით, რომლებშიც ინფორმაცია შეინახება ცალკეულ ჩაკეტილ ბილიკებზე, ოპტიკურ დისკის შემთხვევაში მონაცემები ძველი გრამფირფიტების მსგავსად უწყვეტი სპირალის გასწვრივ ჩაიწერება. გრამფირფიტებისაგან განსხვავებით ოპტიკურ დისკებზე სპირალი იშლება ცენტრიდან ნაპირისაკენ.

დღეს შემდეგი თაობების ოპტიკური დისკები გამოიყენება **DVD** (**D**igital **V**ersatile **D**isk – ციფრული მრავალმიზნობრივი დისკი) და **Blu-ray-დისკები** (**B**lueray– „ციფრული სხივი“). მათი დიამეტრიც სიგრძე ერთხვევა **CD** დისკის დიამეტრიც სიგრძეს, ოღონდ ჩაწერის სიმჭიდროვის ასამაღლებლად მათში გამოიყენება ნაკლები ტალღური სიგრძის მქონე ლაზერი. **DVD** დისკის ტევადობა **17** გბაიტამდეა, **Blu-ray-დისკის** ტევადობა – **500** გბაიტამდე.

დამუშავებულია აგრეთვე კომბინირებული **მაგნიტო-ოპტიკური დისკები**. მათში ინფორმაციის მზიდად გამოყენებულია მაგნიტური ნივთიერება. ლაზერით გახურებისას ეს ნივთიერება ღლევა და მისი ნაწილები ორიენტირდება მაგნიტურ ველში, რაც ცვლის დისკის ზედაპირის ოპტიკურ თვისებებს. ნორმალური ტემპერატურის დაბრუნებისას ასეთი დისკები გარე ზემოქმედებისაღმი საოცრად მდგრადი ხდება. მიუხედავად ამისა, ისინი ჯერჯერობით ვერ გავრცელდა მაღალი ფასისა და ჩაწერის დაბალი სიჩქარის გამო. მაგალითად, ამერიკული კორპორაცია **Maxoptix**-ის მიერ დამუშავებული **T4-2600** ტიპის თანამედროვე მაგნიტო-ოპტიკური დისკი სწრაფმოქმედებით ჩამორჩება ხისტ დიკებს და აჭარბებს ტევადობით, მაგრამ საკმაოდ ძვირია – ღირს **2500** დოლარი.

შევნიშნავთ, რომ ნებისმიერი სახის დისკის სექტორებზე დატანილია ჭდეები, რომელთა წყალობითაც კონტროლერი სწრაფად პოულობს საჭირო ინფორმაციას. თავად მონაცემები მოთავსებულია სექტორის „სათაურსა“ და ამ სექტორის დამამთავრებელ ჩანაწერს შორის.

გარე მეხსიერების მოწყობილობათა სფეროში უკანასკნელ მიღწევას წარმოადგენს **ფლეშმეხსიერების ბაზაზე კონსტრუირებული დამხსობებელი** მოწყობილობა. მასში არ არის მოძრავი ნაწილები, ხოლო ინფორმაციის მზიდს წარმოადგენს ნახევარგამტარული კრისტალი. ფლეშმეხსიერებაში მონაცემების განახლება ბლოკების სახით ხდება, რაც გარე მეხსიერებისათვის სრულიად ბუნებრივია.

თითოეული ბლოკში შემავალი მონაცემების განახლების რაოდენობა საკმოდ დიდია, მაგრამ იგი მანც შეზღუდულია. ამიტომ ფლეშმეხსიერებაში ჩაშენებული კონტროლერი მონაცემების ჩაწერის დროს იყენებს თავისუფალი ბლოკების ამორჩევის სპეციალურ ალგორითმს და ცდილობს, რაც შეიძლება თანაბრად დატვირთოს ბლოკები.

ფართოდ გავრცელებული ფლეშდისკების (ე.წ. „ფლეშკების“) გარდა მეხსიერების ეს სახე გამოიყენება ფოტოაპარატებისათვის, პლემერებისა და მობილური ტელეფონებისათვის განკუთვნილ მეხსიერების ბარათებში, აგრეთვე მყარსხეულიან SSD (Solid State Drive) ვინჩესტერებში.

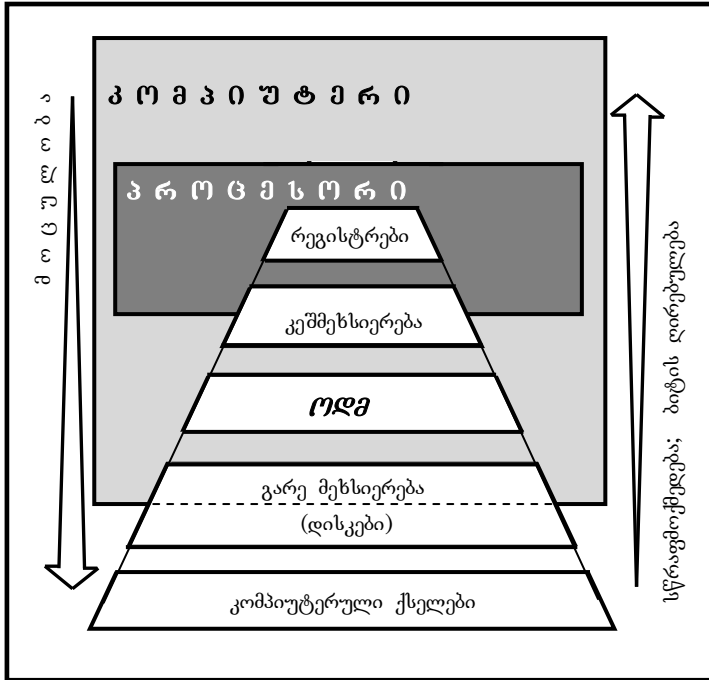
სამეცნიერო ლიტერატურაში 1987 წლიდან შემოვიდა გარე მეხსიერების ახალი სახე, რომლის აღსანიშნავად გამოყენებული იქნა ტერმინი RAID-მასივი (Redundant Array of Independent/Inexpensive Disks – დამოუკიდებელი/იაფი დისკოსატარების ჭარბი მასივი). RAID-მასივის შემოქმედებით ერთიანდება ორი ან მეტი დისკოსატარი და წარმოიქმნება დისკების სისტემა, რომლისთვისაც მაქსიმალურადაა გაზრდილი მუშაობის როგორც სიჩქარე, ისე საიმედოობა; მას აქვს ყველაზე ძვირი დისკოსატარებზე უმჯობესი პარამეტრები, შეუძლია შეინარჩუნოს ინფორმაცია მოწყობილობის ნაწილის მტყუნების დროსაც; ამიტომ თანამედროვე RAID-მასივები აუცილებელია ინფორმაციის მსხვილმაშტაბური საცავების მოსაწყობად.

**სხვადასხვს სახის  
მეხსიერებათა ურთი-  
ერთ შემოქმედება**

1.4 პარაგრაფში აღვნიშნავდით, რომ შეუძლებელია ისეთი მეხსიერების შექმნა, რომელსაც ერთდროულად ექნება დიდი მოცულობა და მაღალი სწრაფმოქმედება (გვ. 31). ამიტომ ვიყენებთ რამდენიმე სხვადასხვა ტიპის მეხსიერებისაგან აგებულ იერარქიულ (მრავალდონიან) სისტემას.

რაც უფრო დიდია მეხსიერების მოცულობა, მით უფრო ნელმოქმედია იგი. სისწრაფის მიხედვით სხვადასხვა სახის მეხსიერებას დონეებად თუ განვალაგებთ, უმაღლეს დონეზე აღმოჩნდება რეგისტრები. სწორედ ისინია ყველაზე სწრაფმოქმედი და, ამასთანავე, ყველაზე მცირე მოცულობის მეხსიერების ტიპური წარმომადგენლები. მომდევნო ქვემოთა დონეს დაიკავებს მათზე გაცილებით დიდი მოცულობისა და, ამასთანავე, მნიშვნელოვნად ნელმოქმედი შიგა

(ოპერატიული დამხსომებელი მოწყობილობა და მუდმივი დამხსომებელი) მეხსიერება. მას მოსდევს ვეებერთელა მოცულობის, მაგრამ გაცილებით უფრო ნელმოქმედი გარე მეხსიერება. და, ბოლოს, უძღაბლესი დონეზეა კომპიუტერული ქსელის მეხსიერება (ნახ. 2.8).



ნახ. 2.8 პერსონალური კომპიუტერის მეხსიერებათა იერარქია

დისკზე (გარე მეხსიერებაში) განთავსებულ ფაილს რედაქტირებისათვის **დამამუშავებელი პროგრამა** ოპერატიულ დამხსომებელ მოწყობილობაში (**ოღმ**-ში) ჩატვირთავს, ხოლო ის კონკრეტული სიმბოლოები, რომლებთანაც პროცესორს წამის უმცირეს მონაკვეთებში უხდება მუშაობა, იერარქიის უმაღლეს დონეზე მდგარ მეხსიერებაში – **რეგისტრებში** თავსდება (იხ. ნახ. 2.8).

კომპიუტერის მწარმოებლურობა, უპირველეს ყოვლისა, მეხსიერების „ზედა“ დონეებზე – პროცესორულ მეხსიერებასა და ოპერატიულ დამხსომებელ მოწყობილობაზეა დამოკიდებული. პროცესორების სწრაფმოქმედება მნიშვნელოვნად აღემატება ოპერატიული დამხს-



ომებელი მოწყობილობის (*ოღმ*-ის) მუშაობის სიჩქარეს, ამიტომ პროცესორს ლოდინი უხდება, სანამ მონაცემები *ოღმ*-იდან მასთან გადავა. სიტუაციის გასაუმჯობესებლად პროცესორსა და *ოღმ*-ს შორის ამატებენ მეხსიერების კიდევ ერთ შრეს, რომელსაც *კემ-მეხსიერებას* ანუ უბრალოდ – *კემს* უწოდებენ (ინგ. *Cashe* – სა-მალაკი, დამალვა).

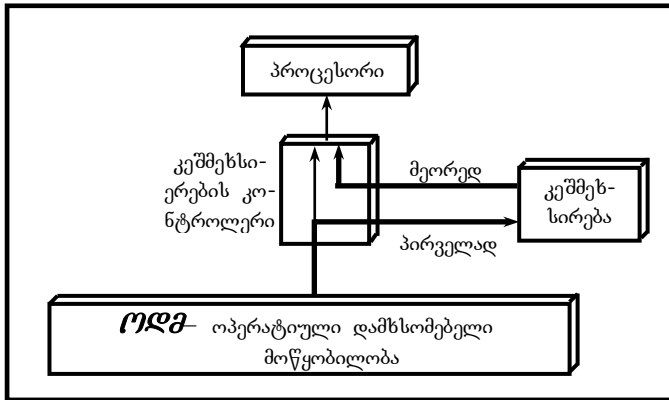
*კემმეხსიერებას*ვა (უფრო ნელმომქმედი) მეხსიერების მუშაობის ამჩქარებელი მეხსიერებაა. ამისათვის იგი ნელმომქმედი მეხსიერებიდან უკვე წაკითხულ მონაცემებს თავისთან ინახავს და საჭიროებისამებრ მათ პროცესორს უსწრაფესად აწვდის; ამით იგი პროცესორს ათავისუფლებს ნელმომქმედ მეხსიერებასთან ხელახალი მიმართვისაგან.

კემმეხსიერება *დინამიკურ* ოპერატიულ დამხსომებელ მოწყობილობაზე გაცილებით სწრაფმომქმედი *სტატიკური მეხსიერებაა*. მას-შიარარისსაკუთარიიმისამართებიდადამისამართებადობისფონ-ნეიმანისეულიპრინციპითარმუშაობს.

*ოღმ*-იდან წაკითხვის დაწყებისას პროცესორი მიმართავს კემ-მეხსიერების კონტროლერს; მასშიშენახულია *ოღმ*-ის ყველა იმ უჯრედის მისამართების სია, რომელთა ასლები კემშია მოთავსებული. მოთხოვნილი მისამართი თუ შედის ამ სიაში, მაშინ პროცესორს *ოღმ*-თან მიმართვა არ სჭირდება, რადგან კონტროლერი მას მიაწვდის ამ მისამართთან დაკავშირებულ (ასოცირებულ) შიგთავსს (ნახ. 2.8). მეხსიერების ორგანიზაციი ასეთ პრინციპს *ასოციაციურ პრინციპს* უწოდებენ.

კემმეხსიერებაში საჭირო მონაცემების არარსებობისას ეს მონაცემები წაკითხება *ოღმ*-დან და ისინი ერთდროულად მიეწოდება როგორც პროცესორს, ისე *კემს* (იხ. ნახ. 2.9). ამ მონაცემებზე ხელახლა მიმართვისას პროცესორი მათ წაკითხავს არა შედარებით ნელმომქმედი *ოღმ*-იდან, არამედ უსწრაფესი *კემმეხსიერებიდან*.

ჩვეულებრივ კემმეხსიერებაში შეიტანება არა მარტო მოთხოვნილი უჯრედის, არამედ მისი უახლოესი უჯრედების შიგთავსები (2.9 ნახაზზე ეს უფრო მსხვილი ისრითაა ნაჩვენები). ამგვარად, კემმეხსიერებაში შენახულია ოპერატიული დამხსომებელი მეხსიერების ყველაზე ხშირად გამოყენებადი უჯრედების შიგთავსები და ისინი პროცესორს ძალიან სწრაფად გადაეცემა.



**ნახ. 2.9** კემქესიერების გამოყენების მაილუსტრირებული სქემა

კემქესიერების მუშაობისას წარმოიშობა შემდეგი ორი ამოცანა:

**1.** კემის მოცულობა ოპერატიული დამსომებელი მოწყობილობის მოცულობაზე გაცილებით მცირეა და იგი სწრაფად ივსება; ამის გამო აუცილებელი ხდება შეიცვალოს ყველაზე „უსარგებლო“ მონაცემები.

**2.** კემქესიერებიდან წაკითხულ მონაცემები თუ პროცესორის მიერ დამუშავდება და ასეთი სახით შეინახება ოპერატიულ მეხსიერებაში, მაშინ უნდა განახლდეს კემის შიგთავსიც.

ორივე ამ ამოცანას წყვეტს კემქესიერების კონტროლერი. სიმნელეების მიუხედავად, კემირება ბევრ შემთხვევაში პროგრამის შესრულების სინქარეს რამდენჯერმე ზრდის.

თავად კემქესიერებაც მრავალდონური სქემით აიგება: თანამედროვე პროცესორებში მას სულ მცირე **2-3** დონე მანც აქვს. ამ დონეებიდან ერთი ნაწილი პროცესორის შემადგენლობაში შედის, ხოლო დანარჩენები განცალკევებული მიკროსქემების სახით აიგება. ამიტომაც **2.8** ნახაზზე ნახვენები კემი ნაწილობრივად რომ არის შესული პროცესორის შემადგენლობაში. განცალკევებულად მზადდება პროგრამებისათვის და მონაცემებისათვის განკუთვნილი კემქესიერება. ეს იმის გამოა მოსახერხებელი, რომ მონაცემებისაგან განსხვავებით, წაკითხული პროგრამა, როგორც წესი, არ იცვლება და ამიტომ მისთვის განკუთვნილი კემი უფრო მარტივად ასაგებია.

გამოთვლით ტექნიკაში ტერმინი „*კეშირება*“ საკმაოდ ფართო მნიშვნელობით გამოიყენება, კერძოდ *კეშირება ეწოდება განმეორებითი გამოყენებისათვის ინფორმაციის შენახვას უფრო სწრაფმოქმედ მექანიზმებში*. მაგალითად, ბრაუზერი ახდენს ინტერნეტიდან მიღებული ფაილების კეშირებას და მათ შეინახავს ხისტ დისკზე არსებულ სპეციალურ საქალაღდეში. ხისტდისკიან დამგროვებელშიც გამოიყენება კეშირება. ასე რომ, კეში შეიძლება რეალიზებული იქნეს, როგორც აპარატურული საშუალებების დახმარებით (ასეთია, მაგალითად *პროცესორის კეში*), ასევე პროგრამულადაც (მაგალითად, *ბრაუზერის კეში*).

რამდენიმე სიტყვით შევეხთ ე. წ. *ვირტუალურ მექანიზმებსაც*. მომხმარებლები და დამპროგრამებლები სხვადასხვა მოთხოვნებს უყენებენ პროგრამულ უზრუნველყოფას. მომხმარებლებს სურთ, რომ პროგრამული უზრუნველყოფა იყოს ინტელექტუალური, ადვილად გამოყენებადი და ითვალისწინებდეს მათთვის საჭირო უმცირეს დეტალებსაც. დამპროგრამებლებს სურთ პროგრამა უმცირეს დროში უმცირესივე ძალისხმევით დაწერონ, ამიტომ ისინი იყენებენ პროგრამების სწრაფად დასამუშავებულ საშუალებებს, ანუ **RAD**-ებს (**Rapid Application Development**). ამის გამო როგორც პროგრამების, ისე დასამუშავებელი მონაცემების მოცულობა სულ უფრო და უფრო იზრდება. ამიტომ კომპიუტერებსაც სულ უფრო და უფრო დიდი მექანიზმება სჭირდება, განსაკუთრებით მრავალამოცანურ რეჟიმის დროს, როდესაც ერთდროულად რამდენიმე პროგრამა მუშაობს.

ოპერატიული დამხსომებელი მოწყობილობის შეზღუდულ მოცულობასთან ზემოთ ჩამოთვლილი მოთხოვნების შესათანხმებლად გამოყენებული იქნა *ვირტუალური მექანიზმების* იდეა. ივარაუდება, რომ კომპიუტერს აქვს ისეთი მაქსიმალურად დასაშვები მოცულობის მექანიზმება, რომელიც საჭიროა პროცესორს მუშაობისათვის; კომპიუტერზე დაყენებულ ოპერატიულ დამხსომებელ მოწყობილობას კი რეალურად ამ სივრცის მხოლოდ ნაწილი უკავია. დანარჩენი ნაწილი განთავსებულია სპეციალურ *სისტემურ ფაილში* ან *ხისტი დისკის* განცალკევებულ განყოფილებაში. ოპერატიული დამხსომებელი მოწყობილობის ფაქტობრივი მოცულობა თუ საკმარისი არ აღმოჩნდება კონკრეტული ამოცანის გადასაწყვეტად, მაშინ სისტემა ოპერატიული მექანიზმებიდან ხისტ დისკზე გადააკოპირებს ის-

ეთ ინფორმაციას, რომელიც მოცემულ მომენტში არ გამოიყენება და ამით ოპერატიულ მეხსიერებაში გამოათავისუფლებს საჭირო მოცულობის არეს. დისკზე გადატანილი მონაცემების საჭიროების გაჩენისას სისტემა მისთვის ანალოგიურად გამოათავისუფლებს ოპერატიული მეხსიერების არეს და მათ იქ დააბრუნებს. სავალდებულო არ არის მონაცემები ოპერატიული მეხსიერების იმავე ადგილზე დაბრუნდეს, რომლიდანაც მოხდა მათი გადატანა (გადაკოპირება).

ვირტუალური მეხსიერების გამოყენებისას ნელდება პროგრამის შესრულება, მაგრამ სამაგიეროდ იგი შეიძლება შესრულდეს არასაკმარისი მოცულობის ოპერატიული დამხსომებლის მოწყობილობიან კომპიუტერზე. დამატებით ოპერატიული მეხსიერების დაყენებისას მრავალჯერ გაიზრდება კომპიუტერის სწარდმოქმედება.

ვირტუალური მეხსიერების გამოყენება კიდევ ერთხელ ადასტურებს იმას, რომ მეხსიერება *ხელოვნურადაა ვაკუოფილი* შიგა და გარე მეხსიერებად. იგი იმითაა განპირობებული, რომ შეუძლებელია ყველა მოთხოვნის ერთდროულად დამაკმაყოფილი იდეალური მეხსიერების აგება.

### მეხსიერების ძირითადი მახასიათებლები

მომხმარებლისათვის, უპირველეს ყოვლისა, მთავარია მეხსიერების ტევადობა, სწრაფმოქმედება და ფასი. *მეხსიერების ტევადობა* იმავე ერთეულებით იზომება, რომლებითაც იზომება ინფორმაციის მოცულობა, ე.ი. ბიტებით, ბაიტებითა და მათგან ნაწარმოები ერთეულებით (უფრო ხშირად, მეგაბაიტებითა და გიგაბაიტებით).

დისკებზე საუბრისას ხშირად ახსენებენ ტერმინებს ფორმატირებული და არაფორმატირებული ტევადობის შესახებ. *დისკის ფორმატირებული ტევადობა* ეწოდება ამ დისკის „სასარგებლო“ ტევადობას, ე.ი. ტევადობას, რომელიც სამომხმარებლო ინფორმაციის შესანახადაა განკუთვნილი; რაც შეეხება *არაფორმატირებული ტევადობას*, იგი სამომხმარებლო ინფორმაციის ჩასაწერად განკუთვნილი ტევადობის გარდა მოიცავს სამომსახურეო მონიშვნებისათვის განკუთვნილ ტევადობასაც.

*მეხსიერების სწრაფმოქმედების* შესაფასებლად რამდენიმე სიდიდეს ვიყენებთ. მონაცემების გაცვლის ნებისმიერი ოპერაცია მოიცავს არა მარტო მონაცემების უშუალოდ გადაცემის პროცესს, არამედ ამ

გადაცემის მოსამზადებლად ჩასატარებელ სამუშაოების შესრულების პროცესსაც. მოსამზადებელ სამუშაოებს მიეკუთვნება, მაგალითად, დისკზე საჭირო სექტორის მოძებნა ან ოპერატიული დამხსომებელი მოწყობილობის მიკროსქემის შივნიტ მისამართის დაყენება. მონაცემების გადაცემის მომზადების დაწყებიდან მონაცემების გადაცემის დამთავრებამდე არსებულ დროის მონაკვეთს **შელწევის (დაშვების) დრო** ეწოდება.

**შელწევის (დაშვების) დრო** ეწოდება საჭირო დროით ინტერვალს ინფორმაციის მოთხოვნის გაგზავნის მომენტიდან მონაცემების სალტზე ამ ინფორმაციის (შელწევის) გაჩენის მომენტამდე.

შელწევის (დაშვების) დროის გაზომვისას ჩვეულებრივ ყველაზე რთულ შემთხვევას განიხილავენ, როდესაც მონაცემები წაიკითხება მეხსიერების შემთხვევითი ადგილებიდან, ან ამ ადგილებზე ჩაიწერება იგი. ბაიტები ან სექტორები ხშირად მიმდევრობით წაიკითხება, ამიტომ მცირდება შეტანის ან გამოტანის დრო.

ოპერატიული დამხსომებელი მოწყობილობებისათვის შელწევის (დაშვების) დრო იზომება **ნანოწამებში** ( $1$  ნანოწამი =  $10^{-9}$  წამს), ხოლო ხისტი დისკებისათვის – **მილიწამებში** ( $1$  მილიწამი =  $10^{-9}$  წამს). ასეთ განსხვავებას განაპირობებს ის, რომ დისკოსატარმა საჭიროა ჯერ მექანიკურად გადაანაცვლოს თავი საჭირო მდებარეობამდე და შემდეგ წაიკითხოს ინფორმაცია.

რადგან გარე მოწყობილობები მონაცემების მთელ ბლოკებს ამუშავებს, ამიტომ მათი დახასიათებისათვის შემდეგი დამატებითი მახასიათებელია საჭირო:

**მონაცემების გადაცემის საშუალო დრო** ეწოდება მოსამზადებელი ეტაპის ჩაუთვლელად, ე. ი. წაკითხვის ოპერაციის უშუალოდ დაწყებიდან დროის ერთეულში გადაცემული მონაცემების რაოდენობას.

ეს მახასიათებელი ჩვეულებრივად იზომება წამში გადაცემული მეგაბაიტებით (მბაიტი/წმ).

დისკური დამგროვებლებისათვის ხშირად მიუთითებენ ბრუნვის სიჩქარეს (წუთში გაკეთებული ბრუნების რაოდენობას). რაც უფრო სწრაფად ბრუნავს დისკი, მით უფრო მაღალი იქნება წაკითხვისა და ჩაწერის დრო.

**მეხსიერების ღირებულების შესაფასებლად** ვიყენებთ მეხსიერების მოდულის ღირებულების ფარდობას მის საინფორმაციო ტევადობას-

თან. ხშირად ვიყენებთ ერთი ბიტის ან ერთი გიგაბაიტის ღირებულების ცნებასაც.

## 2.7. პერსონალურ კომპიუტერში ინფორმაციის შემაჯავებელი მოწყობილობები

პერსონალურ კომპიუტერში იმფორმაციის შესატანად სხვადასხვა სახის მოწყობილობა გამოიყენება, მაგრამ ყველა მათგანს არ ეწოდება შემაჯავებელი მოწყობილობა. მაგალითად, შემაჯავებელი მოწყობილობას არ წარმოადგენს წინა პარაგრაფში განხილული გარე მეხსიერების მოწყობილობები. გლობალური ან ლოკალური ქსელიდან მონაცემების შეტანაც არ მიეკუთვნება *შეტანის პროცესს*, რადგან ამ შემთხვევებში მონაცემები კომპიუტერული ფორმატით უკვე შეტანილი და განთავსებული იყო კომპიუტერულ ქსელში (რომელთანაც მიერთებულია მოცემული პერსონალური კომპიუტერი) და უბრალოდ მოხდა მათი გადაკოპირება.

ხემოთ ჩამოთვლილი ყველა მოწყობილობის გამორიცხვის შემდეგაც მეტადეტად ჭრელი სურათი გვჩნება. ამას ადვილად შევამჩნევთ, თუ ერთმანეთს შევადარებთ კომპიუტერთან მიერთებულ *ტერმინალურების გადაძვინებს*, ინტერნეტში მუდმივად ტრანსლირებადი საქალაქო პეიჯების გადაძვინებს *ვებკამერას*, პროგრამების ასმუშავებლად ან მენიუდან საჭირო ქმედების ამოსარჩევად გამოსაყენებელ *მაუსსა* და, ბოლოს, არა მარტო ტექსტის ასაკრებად, არამედ ბრძანებებს გადასაცემად გამოსაყენებელ *კლავიატურას*.

*შეტანის მოწყობილობა ეწოდება* მოწყობილობას, რომელიც, ჯერ ერთი, მომხმარებელს საშუალებას აძლევს კომპიუტერს გადასცეს ბრძანება და, მეორე, მონაცემებს პირველად გარდაქმნის იმისათვის, რომ მან მიიღოს კომპიუტერში შენახვისა და დამუშავებისათვის საჭირო ფორმა.

შეტანის მოწყობილობებს მიეკუთვნება:

- კლავიატურა;
- მანიპულატორები: მაუსი, ტრეკბოლი, სენსორული პანელი, ჯოისტიკი და ტრეკპოინტი;
- სკანერი;

■ მიკროფონი, ვიდეოკამერა და მულტიმედიური მონაცემების სხვა წყაროები;

■ გრაფიკული ინფორმაციის შეტანის სპეციალიზებული მოწყობილობა - შუქური კალამი და გრაფიკული პლანშეტი;

■ გადამწოდები.

შეტანის ზოგიერთი მოწყობილობა, მაგალითად, კებ-კამერა და გადამწოდები, ადამიანის უშუალოდ ჩაურევლად მუშაობს.

განვიხილოთ შეტანის ზემოთ ჩამოთვლილი მოწყობილობები.

### კლავიატურა (Keyboard)

პერსონალურ კომპიუტერში სიმბოლური ინფორმაციის შემტანი ძირითადი მოწყობილობა კლავიატურაა. კლავიშებზე თითის დაჭრისას შეერთდება კლავიატურის განსაზღვრული ელექტრული კონტაქტები და იმაზე დამოკიდებულებით, თუ რომელ კლავიშზე ან კლავიშების კომბინაციაზე დავაჭირეთ თითი, კომპიუტერის მეხსიერებას გადაეცემა სპეციალური **სკანკოდი** ან ამ კოდების მიმდევრობა.

**სკანკოდი (Scan-code) IBM**-თავსებად კომპიუტერებში თითოეული კლავიშისათვის მიკუთვნებული კოდია, რომლის დახმარებითაც კლავიატურის **დრაივერი** ამოიცნობს დაჭერილ კლავიშს და მის **სკანკოდს** გადასცემს სპეციალურ **60h** პორტში. კლავიშიდან თითის აშვებისას **IBM PC/AT** კომპიუტერის კლავიატურა ზემოთ აღნიშნულ პორტში აგზავნის ორ ბაიტს: **F0h**-სა და **სკანკოდს**. ზოგიერთ (მაგალითად **Shift, Alt, Ctrl** და ა. შ) კლავიშზე თითის დაჭერისას წარმოიშობა არა ერთი, არამედ რამდენიმე სკანკოდი. ისინი ჩაიწერება **60h** პორტში და მოახდენს აპარატურული შეწყვეტის გენერირებას.

არსებობს სხვადასხვა მოდელის კლავიატურები, მაგრამ პრაქტიკულად ყველამათვანში კლავიშები **5** ჯგუფადაა დაჯგუფებული. განვიხილოთ თითოეული მათგანი.

**1. ფუნქციური კლავიშები** აღნიშნულია როგორც **F1-F12** და ისინი კლავიატურის ყველაზე ზემო მწკრივშია განთავსებული. ფუნქციური კლავიშები ძალიან მოსახერხებელია პროგრამებში გამოსაყენებლად, მაგრამ დამწყები მომხმარებლისათვის რამდენადმე ძნელია მათი გამოყენება. ამას განაპირობებს ის გარემოება, რომ აღნიშნული კლავიშების ფუნქციური დანიშნულება თითოეულ კონკრეტულ პრო-

გრამაზეა დამოკიდებული (გამონაკლისია მხოლოდ **F1** კლავიში, რომელსაც პროგრამის მიხედვით გამოაქვს ცნობები). ფუნქციური კლავიშების დაწვრილებით დანიშნულებებს შეგვიძლიათ გავეცნოთ პროგრამის საცნობარო სისტემაში ან დოკუმენტაციაში.

**2. ალფაბეტურიციფრული კლავიშები (48 კლავიში)** ფუნქციური კლავიშების ქვემოთაა განთავსებული. მათზე აღნიშნულია სხვადასხვა ნიშანი, პრობელი, ლათინური და რუსული ასოები. ქართული ასოები შეთავსებულია ლათინურ ასოებთან.

**3. სპეციალური კლავიშები (17 კლავიში)** კლავიატურის სხვადასხვა ადგილზეა განთავსებული. თითოეულ კლავიშს აქვს ზუსტად განსაზღვრული დანიშნულება, კერძოდ:

- **Esc** (ინგ. **Escape** „გასვლა“) კლავიში განკუთვნილია რომელიმე რეჟიმიდან, მოქმედებიდან და პროგრამიდან გამოსასვლელად ან რაიმე მოქმედების გასაუქმებლად;

- **Print Screen/ SysRq** (ინგ. **SystemRequest**) კლავიში ახდენს მიმდინარე გამოსახულების კოპირებას ოპერაციული სისტემის გაცვლის ბუფერში (ეკრანის შიგთავსი გააქვს დასაბეჭდად); კლავიშების **Alt+Print Screen** კომბინაციაზე ერთდროულად ზემოქმედებისას გაცვლის ბუფერში კოპირდება მხოლოდ აქტიური ფანჯარა;

- **Pause/ Break** (*Pause* „შერეობა“, *break* „შეწყვეტა“) კლავიში აჩერებს პროგრამების შესრულებას, მაგალითად, კომპიუტერის ჩატვირთვას;

- **Tab** - ტაბულაციის კლავიში. მასზე გამოსახულია ურთიერთსაპირისპიროდ მიმართული ორი ისარი. ტექსტის შეტანისას იგი ასრულებს აბზაცის ფორმირების ფუნქციას. ზოგიერთ პროგრამაში მას შეიძლება სხვა ფუნქციაც მიენიჭოს;

- **Caps Lock** (ინგ. **Capitals Lock** – „მთავრული ასოების ფიქსირება“) კლავიშზე თითის კენტი რაოდენობით დაჭერის შემდეგ შეგვიძლია ტექსტი მთავრული ასოებით, ხოლო ლუწი რაოდენობით დაჭერისას – ნუსხური ასოებით დავბეჭდოთ;

- **Shift** კლავიშის (კლავიატურაზე არის ორი ასეთი კლავიში) დაჭერილ მდგომარეობაში ბეჭდვისას შეიტანება მთავრული ასოები, ხოლო დაუჭერლად ბეჭდვისას - ნუსხური ასოები;

- **Ctrl** კლავიში (ინგ. **Control**, კლავიატურაზე არის ორი ასეთი კლავიში) გამოიყენება სხვა კლავიშებთან შეხამებულად სხვადასხვა



მოქმედების შესასრულებლად; მაგალითად, **Ctrl + Z** კომბინაცია გამოიყენება ბოლო შესრულებული ქმედების გასაუქმებლად;

- **Alt** კლავიში(ინგ. **Alternate** – შეცვლა, მონაცვლეობა; კლავიატურაზე არის ორი ასეთი კლავიში) გამოიყენება სხვა კლავიშებთან შეხამებულად სხვადასხვა მოქმედების შესასრულებლად; მაგალითად, მენიუს ბრძანების ამოსარჩევად საჭიროა თითო დავაჭიროთ ამ კლავიშს იმ ასოს კლავიშთან ერთად, რომელიც ხაზგასმულია მენიუს დასახელებაში;

- **Backspace** (←; დაბრუნება; „პრობელი უკან“)კლავიში წაშლის კურსორამდე არსებულ ასოს ან ტექსტის გამოყოფილ ბლოკს;

- **Insert** კლავიში გადართავს ტექსტის შეტანის რეჟიმს (ჩასმა ან შეცვლა);

- **Delete** კლავიში გააძევებს კურსორის შემდგომ არსებულ სიმბოლოსან ტექსტის გამოყოფილ ბლოკს;

- **Enter** (ინგ. **Enter** — „შეტან“) ადასტურებს სისტემაში შეტანილ ბრძანებას ან ტექსტის აკრეფის დროს აფიქსირებს აბზაცის დამთავრებას;

- **NumLock**(პირდაპირი თარგმანი „ციფრების ფიქსირება“) კლავიში ასრულებს გარკვეულ ბლოკირებას, რომელიც აუცილებელია კლავიატურის დამატებით ნაწილში არსებული კლავიშების მუშეობით ციფრების შესატანად.

**4. სანავიგაციო კლავიშები (8 კლავიში)** განთავსებულია ალფაბეტურციფრული კლავიშების მარჯვნივ და მართავს კურსორის მდებარეობას; კერძოდ ←; ↑; → და ↓ ისრებიანი კლავიშებიკურსორს ერთი ბიჯით გადაადგილებს შესაბამისად მარცხნივ, ზემოთ,მარჯვნივ და ქვემოთ; **Home**და **End** კლავიშებს კურსორი შესაბამი-სად გადააქვს ტექსტის თავსა და ბოლოში, ხოლო **Page Up** და **Pa-ge Down** კლავიშებს – ფურცლის ზემოთ და ქვემოთ.

**6. დამატებითი კლავიშები (16 კლავიში)** განთავსებულია კლავიატურის მარჯვენა ნაწილში. ისინა ახდენს სანავიგაციო და ციფრული კლავიშების დუბლირებას, შეიცავს მათემატიკური /, \* - ; +); მოქმედებების ნიშნით აღნიშნულ კლავიშებსა და კლავიშებს **Insert**, **Delete**, **Enter**.

ოპერაციული სისტემა **Windows 95**-ის გამოსავლის შემდეგ კლავიატურაზე გაჩნდა სამი **Windows**-კლავიშისაგან შემდგარი

ჯგუფი. **WL** (**Windows Logo**) ლოგოტიპიანი ორი კლავიში კლავიატურის ორივე მხარეზე, ხოლო **WA** (**Windows Application**) ლოგოტიპიანი კლავიში – კლავიატურის მხოლოდ მარჯვენა მხარეზე განთავსებული. **WL** კლავიშის მეშვეობით შეგვიძლია გამოვიძახოთ სისტემის მთავარი მენიუ, ხოლო **WA** კლავიშის მეშვეობით – კონტექსტური მენიუ.

**Windows 98**-ის გამოსავლის შემდეგ კლავიატურაზე გაჩნდა სამი კლავიშისაგან შემდგარი კიდევ ერთი ჯგუფი; მასში შემავალ კლავიშებს *ძილის კლავიშებს* ეწოდებენ. **Power off** კლავიში ამორთავს კომპიუტერს, **Sleep** კლავიში, რომელზეც მთვარეა გამოსახული, აძინებს, ხოლო **WakeUp** კლავიში, რომელზედაც მალვიძარა საათია გამოსახული, აღვიძებს კომპიუტერს.

**ACPI** (**A**dvanced **C**onfiguration and **P**ower **I**nterface – კონფიგურაციისა და კვების ვაფართოებული ინტერფეისი) სპეციფიკაციის თანახმად ახალმა **Windows 98**-მა „ისწავლა“ განრიგის ან გარკვეული მოვლენის მონდენის (კლავიშზე თითის დაჭერის, მაუსის გადაადგილების) შესაბამისად კომპიუტერის კომპონენტების კვების მართვა; კერძოდ, მას შეუძლია „ღრმად დააძინოს“ (**Standby**) ან გააღვიძოს ისინი.

ზოგიერთი სახის კლავიატურაზე არის **Fn** (ინგ. **F**unction) კლავიში, რომელიც აფართოებს კლავიატურის ფუნქციურ შესაძლებლობებს; იგი სხვა, მაგალითად, *ძილის კლავიშებთან*, ერთად გამოიყენება, რაც ნულამდე ამცირებს მათზე შემთხვევით დაჭერის ალბათობას. ჩვეულებრივ იგი მარჯვენა ზედა კუთხეში განთავსდება კლავიშ **Enter**-ის ქვემოთ.

ბოლო წლებში ფართოდ ვრცელდება *მულტიმედია კლავიატურა*, რომელზეც განთავსებულია კომპიუტერის სპეციალური რეჟიმების მმართველი (დისკისა და საკვრელი, ხმის ამომრთველი, კომპაქტდისკის ამამუშავებელი და სხვ.) კლავიშები.

ინტერნეტის ქსელის ფართოდ განვითარებამ განაპირობა კლავიატურაზე შესაბამისი ღილაკების გაჩენა, რომლებიც საშუალებას გვაძლევს მივუერთდეთ ქსელს, გავიდეთ საიტზე, მივიღოთ ფოსტა და ა. შ. ჩვეულებრივად ასეთი ღილაკები სულ ზემოთ, ფუნქციური კლავიშების თავზე თავსდება. ხშირად ახალ კლავიატურებში ვხვდებით სპეციფიკური ფუნქციების შემსრულებელ დამატებით ღილაკებსაც.

არსებობს სადენიანი და უსადენო (*Wireless; Cordless*) კლავიატურები. უკანასკნელებს არ სჭირდება სისტემურ ბლოკთან საკაბელო მიერთება (უფრო ზუსტად, მათ შეუძლია მიუერთებლადაც იმუშაოს). ყველა კლავიატურა იყენებს ქვემოთ ჩამოთვლილი **3** ინტერფეისიდან ერთ-ერთს:

**1. AT (Advanced Technology) ინტერფეისი** გამოიყენება AT კვების მქონე სისტემური დაფებისათვის. იგი წარმოადგენს სტანდარტიზაციის გერმანული ინსტიტუტის მიერ დამუშავებულ სქელ **5**-კონტაქტიან *DIN* (გერმ. *Deutsches Institut für Normung*) გასართოს, რომელიც ძველ ელექტრონიკაში გამოიყენებოდა დაბალსიხშირული აუდიო- და ვიდეოსიგნალების გადასაცემად;

**2. PX/2 ინტერფეისი** გამოიყენება ATX (იხლ *Advanced Technology Extended* გაფართოებული მოწინავე ტექნოლოგია) კვების მქონე სისტემური დაფებისათვის. იგი წარმოადგენს წვრილ მრგვალ **6**-კონტაქტიან *miniDIN* გასართოს. იგივე გასართი გამოიყენება მაუსისთვისაც. *PC'99* სპეციფიკაციაში ამ შტეკერებს სხვადასხვა ფერად (იისფერად – კლავიატურისათვის და მწვანედ – მაუსისათვის) დებავენ.

**3. USB ინტერფეისი** გამოიყენება მეტ-ნაკლებად თანამედროვე სისტემურ დაფებზე. იგი ბრტყელი მართკუთხეიანი **4**-კონტაქტური გასართია.

### მანიპულატორები

კომპიუტერში ბრძანებებისა და მონაცემების შესატანად ფართოდ გამოიყენება მანიპულატორები – სხვადასხვა კონსტრუქციის მოწყობილობა, რომლებზედაც სხვადასხვა სახის ზემოქმედებით მომხმარებელს შეუძლია ტექსტის აუკრეფად მართოს კომპიუტერი.

მანიპულატორების მუშაობის ზოგადი იდეა ასეთია. მონიტორის ეკრანზე აისახება მაჩვენებელი (*კურსორი*); ხელით მანიპულატორის გადანაცვლების მეშვეობით კურსორი შეგვიძლია ვამოძრაოთ ეკრანზე. საჭირო ადგილზე მისი მისვლის ფაქტი მანიპულატორზე არსებულ ღილაკზე თითის დაჭერით შეგვიძლია ვაცნობოთ კომპიუტერს. ეკრანზე თუ კლავიატურას გამოვსახავთ, მაშინ მანიპულატორის მეშვეობით ტექსტის კრეფასაც შევძლებთ.

ყველაზე გავრცელებული მანიპულატორია *კომპიუტერული მაუსი (თავი)* (ნახ. **2.10,ა**). ეს სახელი დაკავშირებულია კომპიუტერთან მოწყობილობის შემაერთებელ კაბელთან, რომელიც თავის

*კულის ასოციაციას* აღძრავს. მრავალ თანამედროვე მაუსს უკვე აღნიშნული ვირტუალური კუდი სრულებით არ სჭირდება, რადგან კომპიუტერს საჭირო ინფორმაციას ისინი ელექტრომაგნიტური ტალღებითაც მშვენივრად გადასცემს. ამ დროს კომპიუტერთან მიერთებული უნდა იყოს რადიოტალღების აღქმელი და გამშიფრავი სპეციალური მოწყობილობა – *ადატერი*. ასეთი მაუსების გამოყენება ძალიან მოსახერხებელია, ოღონდ ძვირია და კვების დამატებით წყაროს (ბატარეას ან აკუმულატორს) იყენებს. პირველი მაუსის დემონსტრირება **1964** წელს მოახდინა სტენფორდის კვლევითი ცენტრის თანამშრომელმა *დუგლას ენლბარტმა*. **1983** წელს ფირმა *Microsoft*-მა *IBM PC* კომპიუტერისათვის გამოუშვა *Bus Mouse*-ს სახელწოდების პირველი მაუსი, ხოლო **1986** წელს გამოჩნდა ახალი მაუსი *InPort Mouse*.

თავდაპირველად მაუსი აღჭურვილი იყო მოძრაობის მექანიკური გადაამწოდებით: მაუსის გადანაცვლებისას ბრუნვას იწყებდა მაუსის შიგნით არსებული ბურთულა. ბურთულა თავის მხრივ აბრუნებდა ერთმანეთისადმი ურთიერთმართობულად განთავსებულ ორ პატარა ბორბალს, რომელთა ყოველ მობუნებას აფიქსირებდა ელექტრონული მოწყობილობა. ეს ბორბლები შეიძლება ჰიპოთეტურ კოორდინატთა ღერძებს, ხოლო მათი მობრუნება - კოორდინატების ცვლილებას შევადაროთ. კოორდინატების ცვლილების შესახებ ინფორმაცია გადაეცემოდა კომპიუტერს. ასეთი მექანიკური კონსტრუქცია მოუხერხებელი იყო, რადგან მტვერისა და ჭუჭყისაგან ბურთულისა და პატარა ბორბლების ხშირად გაწმენდა გვიხდებოდა.

დღეს გავრცელებული *ოპტიკური მაუსები* არ შეიცავს მექანიკურ ნაწილებს, რაც ამაღლებს მათ ხანგამძლეობასა და სიზუსტეს. მაუსის „მუცლის“ ქვეშ განთავსებული მინიატურული ვიდეოკამერა დროის ყოველ მცირე შუალედებში სურათს უღებს მაუსის შუქლიდით ან პორტატული ლაზერით გაშუქებულ იმ ზედაპირს, რომელზედაც იგი მოძრაობს. ამ სურათების ურთიერთშედარებით მაუსის „სხეულში“ ჩამონტაჟებული სპეციალური მიკროპროცესორი გამოითვლის ორ ღერძს შორის მაუსის გადაადგილებას. ეს მეთოდი ცუდ შედეგს მოგვცემს, თუ ზედაპირი ძალიან გლუვი და ერთგვაროვანია (როგორცაა, მაგალითად, მიწის ზედაპირი). ასეთ შემთხ-

ვევაში გამოიყენება **ლაზერული მაუსები**, რომლის გაშუქება ზრდის გამოსახულების კონტრასტულობას.

ოპტიკური მაუსის ერთ-ერთი უმნიშვნელოვანესი მახასიათებელია **ოპტიკური სენსორის (ვიდეოკამერის) განრჩევადობა**. იგი გვიჩვენებს მოცემული სიგრძის მონაკვეთს თუ რამდენ წერტილს განასხვავებს ერთმანეთისაგან მოცემული სენსორი (ვიდეოკამერა), ე.ი. რამდენი წერტილის აღქმა შეუძლია მას. რაც უფრო მაღალია სენსორის განრჩევადობა, მაუსი მით უფრო ზუსტად მიაღწევს თვალს გადაადგილებას. ეს მნიშვნელოვანია, მაგალითად, გრაფიკულ რედაქტორში, გამოსახულების ზუსტად დამუშავებისთვის. განრჩევადობა იზომება დუიუმებში ანუ **dpi**-ში (**dots per inch** – წერტილები დუიმზე). მაუსის ჩვეულებრივი განრჩევადობაა **1000dpi**, ხოლო განსაკუთრებულად ზუსტ მაუსებს რამდენჯერმე მეტი განრჩევადობა აქვს.

გარდა განრჩევადობისა, მაუსის მუშაობის ხარისხზე გავლენას ახდენს ის, თუ რამდენი კადრის გადაღება შეუძლია ვიდეოკამერას ერთ წამში. ჩვეულებრივ მაუსს წამში **1000**-მდე კადრის გადაღება შეუძლია. თითოეული კადრის ზომები გადაწოდზეა დამოკიდებული და იგი (**16 × 16**) პიქსელიდან - (**30 × 30**) პიქსელამდე იცვლება. ამ მონაცემების ცოდნისას შეგვიძლება ზუსტად გამოვთვალოთ **მაუსის სწრაფმოქმედება**, ე.ი. წამში თუ რამდენი მეგაპიქსელის გამოსახავს იგი (მპ/წმ). კომპიუტერული თამაშებისათვის გამოსაყენებელი მაუსისათვის მნიშვნელოვანია **მოძრაობის მაქსიმალური სიჩქარე** – მან წამში შეიძლება რამდენიმე მეტრს მიაღწიოს.

**მაუსები** ერთმანეთისაგან განსხვავდება საკუთარი მმართველი შესაძლებლობითაც – ადრე არსებობდა ძირითადად ორლილაკიანი და სამლილაკიანი **მაუსები**. სამლილაკიან მაუსებს ახლა იშვიათად ვხვდებით, რადგან მართვისათვის ორი ღილაკი სრულიად საკმარისია. ოპერაციების უმრავლესობა სრულდება მარცხენა ღილაკით, რომელიც უპირობოდაა აქტიური (**Windows**-ში შეიძლება შეიცვალოს მაუსის ღილაკების აწყობა).

**მარცხენა ღილაკი** განკუთვნილია:

■ ინფორმაციის შესატანად (მაუსის მაჩვენებელი უნდა დავაყენოთ მართვის ობიექტზე და თითო ერთხელ დავაჭიროთ მარცხენა ღილაკს);



*ნახ.2.10. პერსონალური კომპიუტერში გამოყენებული მანიპულატორები*

■ გარკვეული ობიექტის გამოსაყოფად ან ხატვისათვის (მარცხენა ლილაკს უნდა დავაჭიროთ თითი და მის აუშვებლად მაუსი გადავადგილოთ საჭირო ადგილამდე); გამოყოფის შემდეგ თითი უნდა ავუშვათ მარცხენა ლილაკს, ე. ი. გამოვიყენოთ პრინციპი: „*დააჭირე, გადაათრე, დააგდე*“.

**მარჯვენა ღილაკი** განკუთვნილია სამუშაოდ ამორჩეული ობიექტის კონტექსტური მენიუს გამოსაძახებლად, აგრეთვე საჭირო ობიექტის ეკრანზე გადასადგილებლად (კოპირებისათვის). ბოლო წლებში ფართოდ გამოიყენება **მაუსი**, რომელსაც **სკორლინვისათვის** აქვს სპეციალური სატრიალო თვალი. **სკორლინგი** – ეს არის გამოსახულების, ტექსტის ან **verb**-გვერდის „დატრიალება“ ვერტიკალურად ან ჰორიზონტალურად. სატრიალებელი თვლიანი მაუსი ძალიან მოსახერხებელია დიდ ტექსტებთან, ცხრილებთან, სურათებთან, ინტერნეტის დოკუმენტებთან მუშაობისას.

■ ბურთულაკიანი მანიპულატორი – **ტრეკბოლი** (ნახ. 2.10,ბ) შეიძლება გადმობრუნებულ მაუსს შევადაროთ. მის მგრძნობიარე ელემენტად თავისი ღერძის ირგვლივ მბრუნავი ჩამაგრებული ბურთულა გამოიყენება. სახელწოდება „ტრეკბოლი“ წარმოშობილია ინგლისური სიტყვებისაგან **track**, რომელიც **მიმმართველ მოწყობილობას** და **ball**, რომელიც **სფეროს**, **ბურთის** ნიშნავს. ტრეკბოლი არ საჭიროებს სწორ ზედაპირს და მას ვიბრაციის დროსაც შეუძლია მუშაობა. დღეს **ტრეკბოლი** პრაქტიკულად არ გამოიყენება.

■ **ნოუთბუკებში** მაუსის ჩაშენებულ „შემცვლელად“ აყენებენ კიდევ ერთი სახის მანიპულატორს – **სენსორულ პანელს** (ინგ. **Touchpad**). პანელი შედგება დაჭერისადმი მგრძნობიარე მცირე ზედაპირისა და ორი ღილაკისაგან. მგრძნობიარე პანელზე შეხება მაუსის ღილაკზე თითის დაჭერის ტოლფასია (შეიძლება პანელის გვერდზე არსებული ღილაკითაც ვისარგებლოთ). ზოგიერთ პანელს არა მართო გადახანაცვლების, არამედ სხვა ბრძანებების შესრულების უნარიც აქვს. მაგალითად, დოკუმენტის „დასატრიალებლად“ შეიძლება პანელის მარჯვენა და ქვედა საზღვრებთან (იქ, სადაც მიღებულია ფანჯარაში განთავსდეს „დამტრიალებელი“ ზოლი) თითი გავუსვათ. ზოგიერთ პანელს რამდენიმე წერტილში შეხების გაანალიზების უნარიც აქვს. ამას **მულტიტაჩის (Multitouch** – მრავლობითი შეხება) რეჟიმი ჰქვია.

■ **ჯოისტიკი** (ნახ. 2.10, გ) „ნაკლებად სერიოზული“ მანიპულატორია; მისი სახელწოდება ინგლისური „**joy stick**“-ის აბრევიატურაა და ქართულად შეიძლება ვთარგმნოთ როგორც „მხიარული“ სახელური. იგი ძირითადად კომპიუტერული თამაშებისათვის გამოიყენ-

ნება და შეიძლება მეტად უცნაური ფორები ჰქონდეს. ჯოისტიკს აქვს სახელური, რომლის მოტრიალების შესაბამისად კორპუსის შიგნით შეირთვება კონტაქტები. ზოგიერთ მოდელში დამატებით დაყენებულია წნევის გადამწოდი და მომხმარებელი რაც უფრო ძლიერ გადახრის სახელურს, მით უფრო სწრაფად მოძრაობს კურსორი ეკრანზე.

■ **ტრეკპოინტი**(ნახ. 2.10, დ) ინგლისური სიტყვაა და ქართულად კურსის ანუ მარშრუტის მაჩვენებელს ნიშნავს. იგი არის ნეტბუკსა (მცირე ზომის ნოუთბუკსა) და ზოგიერთ ნოუთბუკში მაუსის ნაცვლად გამოყენებული მინიჯოისტიკი. ტრეკპოინტი პირველად 1992 წელს გამოიყენა კომპანიამ IBM. ასეთი მოწყობილობა განთავსებულია კლავიატურის შუა ნაწილში G, H და B ასოებს შორის, ხოლო მაუსის დილაკების შემცვლელი მარცხენა და მარჯვენა დილაკები პრობელის კლავის ქვემოთაა მოთავსებული.

### სკანერი

**სკანერი** (ნახ. 2.11, ა) ეწოდება ბრტყელ ზედაპირზე დატანილი გამოსახულების შემტან მოწყობილობას. მისი დახმარებით შეგვიძლია კომპიუტერულ მონაცემებად გარდავქმნათ (ციფრული ფორმით წარმოვადგინოთ) სიმბოლური (ტექსტური და ციფრული) ინფორმაცია, ნახატები, ფოტოსურათები, ფოტოლენტაზე არსებული გამოსახულებები (ნეგატივები და სლაიდები); გარდა ამისა, მისი საშუალებით შეგვიძლია მივიღოთ მცირე სისქის ობიექტების ფოტოსურათები.

ხშირად სკანერის დახმარებით კომპიუტერში შეაქვთ საოფისე დოკუმენტები. ამ დროს სკანერი კომპიუტერს შესატან დოკუმენტს ფოტოსურათის სახით გადასცემს. გადასასკანირებელი ტექსტის რედაქტირებისათვის საჭიროა ეს სურათი სიმბოლოებისაგან შემდგარ კოდებად გარდაიქმნას. ამისათვის გამოიყენება **სიმბოლოების ოპტიკური შეცნობის OCR- პროგრამები** (ინგ. **Optical Character Recognition**). **OCR-პროგრამები** ცდილობს ნებისმიერი ასოს მოხაზულობის სურათი პიქსელების ერთობლიობის სახით დააფიქსიროს და კომპიუტერში არსებულ ასოების ნიმუშებთან მისი შედარების გზით განსაზღვროს, თუ რომელ ასოს წარმოადგენს ეს დაფიქსირებული ერთობლიობა. პრინციპულად შესაძლებელია ხელნაწერი ტექსტის შეცნობაც, მაგრამ ამ ამოცანას პროგრამები ნაბეჭდი ტექსტის შე-



ცნობის ამოცანაზე უფრო ცუდად წყვეტს (ხელნაწერი ასოების არაერთგვაროვნობის გამო).

სკანერის მუშაობის პრინციპი ძალიან მარტივია. მოკაშკაშე წყაროდან წარმოქმნილი სხივი გადაივლის გადასასკანირებელ ზედაპირზე (ინგლისური სიტყვა *Scan* დაფინებულად დათვალიერება-შესწავლასაც და თვალის გადავლებასაც ნიშნავს), რომლის დროსაც შუქმგრძობიარე გადამწოდები აღიქვამს არეკლილ სხივებს და განსაზღვრავს მათ როგორც ინტენსიურობასა და ფერს. შეიძლება ვთქვათ, რომ სკანერი ძალზე გამარტივებული ციფრული ფოტოაპარატია. არსებობს შემდეგი კონსტრუქციის სკანერები.

■ **ხელის სკანერი**, რომელშიც შუქმგრძობიარე თავს მომხმარებელი გადასასკანირებელ ზედაპირზე ისე გადაატარებს, როგორც მაღაზიაში გამყიდველი გადაატარებს წამკითხავ თავს საქონელზე არსებულ შტრიხკოდზე;

■ **პლანშეტური სკანერი**, რომელშიც გადასასკანირებელ ობიექტს მომხმარებელი უძრავად დებს მინაზე, ხოლო სკანერის შიგნით არსებული შუქმგრძობიარე თავი თვითონ გადაუვლის ამ ობიექტს;

■ **რულონური სკანერი**, რომელიც რულონად დახვეულ გამოსახულებიან ქაღალდს თვითონ გაჭიმავს უძრავი შუქმგრძობიარე თავის გავლით;

■ **დოკუმენტი სკანერი**, სადაც გადასასკანირებელი ობიექტი უნდა დავაწებოთ უძრავი თავის გვერდით მბრუნავ დოკუმენტზე. ასეთი სკანერი უზრუნველყოფს სკანირების საუკეთესო ხარისხს და საგამომცემლო საქმეში გამოიყენება.

სკანერის შერჩევისას საჭიროა ყურადღება მივაქციოთ შემდეგ ექვს მახასიათებელს:

**1. ოპტიკურ და ინტერპოლაციურ განრჩევადობას**, რომელიც, როგორც ზემოთ ავლინებთ, ერთ დუიმზე აღქმული წერტილების რაოდენობით გაიზომება (**Dots Per Inch - dpi**). **ოპტიკური განრჩევადობა** სკანერის ერთ-ერთი მთავარი ტექნიკური მახასიათებელია. იგი გვიჩვენებს სურათის სიმაღლესა (ვერტიკალურად) და სიგრძეზე (ჰორიზონტალურად) განთავსებული პიქსელების (მნათი წერტილების) რაოდენობას და ჩაიწერება, მაგალითად, ასე: **600×1200**. რაც უფრო მაღალია ეს მაჩვენებელი, მით უფრო მაღალია ორიგინალიდან აღქმული ინფორმაცია. **ინტერპოლაციური განრჩევადობა** – სპე-



**ნახ.2.11.** პერსონალური კომპიუტერში ინფორმაციის შეტანისა და გამოტანის ზოგიერთი მოწყობილობა

ციალური ალგორითმით დრაივერის მიერ უკმარი წერტილების რაოდენობის მათემატიკური გამოთვლის (ანუ, ინტერპოლაციის მეთოდის გამოყენების) გზით განსაზღვრული ხელოვნურად გაზრდილი განრჩევადობაა. ეს პარამეტრი მწარმოებლებს ხშირად შეაქვთ სკა-

ნერების დასახელებაში, რათა გაუმჯობესდეს სკანერის მახასიათებლები.

**2. ფერის სიღრმეს,** რომელიც ასახავს თუ რამდენ შეფერილობას აღიქვამს მასკანირებელი მატრიცა. თითოეული შეფერილობა თუ **24** ბიტური კოდით კოდირება, მაშინ მასკანირებელი მატრიცა შეძლებს, აღიქვას  $2^{24} \approx 16,8$  მილიონი შეფერილობა, ხოლო თუ ამისათვის იყენებს **48** ბიტს, მაშინ აღქმული შეფერილობების რაოდენობა  $2^{48} \approx 281,5$  ტრილიონის ტოლი იქნება. საინტერესოა აღვნიშნოთ, რომ ადამიანის თვალს ერთდროულად **24** ბიტური ფერის აღქმაც არ შეუძლია.

**3. მუშა ზონის ზომას,** რომელიც განსაზღვრავს სკანირების მაქსიმალურ ზონას. **A4 (210×297)** მმ ფორმატის პლანშეტური სკანირების მუშა ზონა იცვლება (**210×297**) მილიმეტრიდან (**216×355**) მილიმეტრამდე;

**4. ინტერფეისს,** რომელიც განსაზღვრავს კომპიუტერში მონაცემების გადაცემის სიჩქარეს. თავდაპირველად ყველაზე გავრცელებული იყო ჩვეულებრივი პარალელური **LPT** ინტერფეისიანი (ინგ. **Line Print Terminal – სტრიქონის ბეჭდვის ტერმინალი**) მოდელები, რომლებსაც **SCSI**–მოდელი (ინგ. **Small Computer System Interface – მცირე კომპიუტერული სისტემებს ინტერფეისი**) ეწოდებოდა. ახლა ფართოდ გავრცელდა **USB 2,0** ინტერფეისი;

**5. სკანირების მეთოდს,** რომელიც გამოსახულებაზე სკანერის გადავლის რაოდენობას განსაზღვრავს (პლანშეტური სკანირების უმრავლესობა გამოსახულებაზე ერთხელ გადაივლის);

**6. სკანირების სიჩქარეს,** რომელიც დამოკიდებულია მოცემულ განრჩევადობაზე, სასკანერო გამოსახულების ზომაზე, ინტერფეისისა და სკანირების მეთოდზე. სკანირების სიჩქარე ამავე დროს დიდად კომპიუტერის სწრაფმოქმედებაზედაცაა დამოკიდებული.

სკანერს, გრაფიკულ პროგრამასა (გამოყენებასა) და მომხმარებელს შორის ურთიერთობის საშუალებაა **twain**-დრაივერად წოდებული სპეციალური პროგრამა (ოფიციალურად **twain** არ წარმოადგენს აბრევიატურას, თუმცა გავრცელებულია ვერსია: **Technology-Without Any Interesting Name**– რაიმე ნიშნადი დასახელების არმქონე ტექნოლოგია). იგი ასრულებს სკანერის მართვის ფუნქციას, წინასწარი გასინჯვის რეჟიმში გამოსახულების კორექტირების საშუალებას გვაძლევს და მონაცემებს

სკანერიდან გადასცემს კომპიუტერში. პროგრამა საშუალებას გვაძლევს, დავეყნოთ სკანირების ზონის პარამეტრები, მოვახდინოთ წინასწარი სკანირება და გასინჯვა, უზრუნველყოფს ფერის კორექტირებასა და მიღებული გამოსახულების დაშუშავებას. სკანერის თითოეული მწარმოებელი თავად ქმნის საკუთარ *twain*-დრაივერს. ნებისმიერ მათგანს პრაქტიკულად აქვს კონტრასტულობის რეგულირების, *გამა-კორექციის* უნარი და მომხმარებელს ნებისმიერი მათგანი აძლევს განრჩევალობის, სიკაშკაშის რეგულირებისა და მონიტორის ეკრანზე *მუარის* აღმოფხვრაში შესაძლებლობას.

*ზემოთ მოყვანილი მსჯელობაში გასარკვევად შევნიშნავთ:*

**1. გამა კორექცია** ეწოდება გამოტანის მოწყობილობის მახასიათებლებზე დამოკიდებულებით მონიტორის ეკრანის სიკაშკაშის კორექტირებას;

**2. მუარი** ეწოდება მონიტორის ეკრანზე ინტერფერენციული მუქი ვერტიკალური ნახევრადწრისებრი ხაზების გაჩენას და არ წარმოადგენს მონიტორის უწყისეობას. მოცემული ვიზუალური ეფექტი მეტ-ნაკლებლად ყველა მონიტორზეა. ამ პარაზიტული ეფექტის შესამცირებლად საჭიროა ისე შევარჩიოთ განრჩევალობა და გაშლის სიხშირე, რომ მუარი ნაკლებად შესამჩნევი გახდეს. ხშირად მუარის რეგულირება მომხმარებლის ფუნქციაა, მაგრამ მონიტორების ზოგიერთ ძვირად ღირებულ მოდელებში ჩაშენებულია მუარის მარეგულირებელი სისტემა.

ძვირადღირებულ მოწყობილობებს მიეკუთვნება **სამგანზომილებიანი სკანერები**. ძალიან რთული და შრომატევადია მოცულობითი ვიზუალიზაციის პროცესი. კონსტრუქტორები უკვე იყენებენ სამგანზომილებიან **დიგიტალიზერებს** (ინფორმაციის ხელით უშუალოდკომპიუტერში შემტან მოწყობილობებს, რომლებიც შედგება ბრტყელი სენსორული პლანშეტისა და კალმისაგან), რომელთა საშუალებითაც შემდგომი ვირტუალური დამუშავებისათვის კომპიუტერში შეგვიძლია შევიტანოთ რეალურად არსებული საგნების სამგანზომილებიანი მოდელები. აქტიური სამგანზომილებიანი სკანერი იყენებს დასამუშავებელი ობიექტიდან არეკლილ ლაზერულ სხივებსა და დღის მუქის განმზნევი სპეციალური ფილტრებით აღჭურვილ შესაბამის სენსორს (წაშში **2400** კადრის გადამღებ ვიდეოკამერას).

## 2.8. პერსონალური კომპიუტერიდან ინფორმაციის გამოტანის მოწყობილობები

*გამომტანი მოწყობილობა* ეწოდება მოწყობილობას, რომელიც კომპიუტერულ მონაცემებს ადამიანის (მომხმარებლის) აღსაქმელად მოსახერხებელ ფორმას აძლევს. მათი საშუალებით ინფორმაცია კომპიუტერიდან ადამიანს გადაეცემა.

გამომტანის ძირითადი მოწყობილობებია მონიტორები, პრინტერები, სპიკერები, ყურსასმენები, პლოტერები (გრაფომზიდები).

### მონიტორები

მონიტორი ეწოდება ტექსტური და გრაფიკული ინფორმაციის ვიზუალურად ამსახავ (გამომტან) მოწყობილობას.

დღეს ინფორმაციის ყველაზე მეტად გავრცელებულია თხევადკრისტალური **LCD**-მონიტორები (ინგ. **Liquid Crystal Display, LCD**). არსებობს იშვიათად გამოყენებადი **CRT**- და **GPDP**-მონიტორებიც. **CRT**-მონიტორების (ინგ. **Cathode Ray Tube, LCD**) გამოყენების იშვიათობას განაპირობებს ის ფაქტი, რომ ისინი უკვე მოძველდა და ტექნიკური პარამეტრებით ჩამოუვარდება **LCD**-მონიტორებს, ე.ი. ისინი გუშინდელი დღის მონიტორებია. რაც შეეხება **GPDP**-მონიტორებს (ინგ. **Gas-Plasma Display Panel, GPDP**), მათი გამოყენების იშვიათობა განპირობებულია სიძვირისა და ზოგიერთი ტექნიკური სიერთულის გამო; ტექნოლოგიების სათანადოდ დახვეწის შემთხვევაში არაა გამორიცხული, რომ ისინი ძირითად მონიტორებად გადაიქცეს.

■ **მილაკური CRT-მონიტორები** (ნახ. 2.11,ბ) აგებულია კათოდურსხივიური, ანუ ელექტრონულსხივიური მილაკის გამოყენებით. ამ მილაკის შიგნით ვაკუუმი, ხოლო წინა მხარის შიგა ნაწილი ლუმინოფორის შრითაა დაფარული, რომელიც სამი ელექტრონული სამი სატყორცნიდან დაშენილი დამუხტული ნაწილაკებით ინთება. ლუმინოფორის ზედდება ხდება სამი ძირითადი (წითელი - **Red**, მწვანის - **Green** და ლურჯი - **Blu**) ფერის წერტილების ერთობლიობით. ამ ფერების ურთიერთშეხამებით მიიღება 16 მილიონამდე რაოდენობის სხვადასხვა ფერი, რომელთა მხარდაჭერა შეუძლია თანამედროვე ვიდეობარათს.

ლუმინოფორის წერტილების ნაკრებები სამკუთხოვანი ტრიადების სახითაა განთავსებული. ასეთ ტრიადას პიქსელი ეწოდება. პიქ-

სელებისაგან მონიტორზე ფორმირდება გამოსახულება. პიქსელების ცენტრებს შორის დიაგონალურად არსებულ მანძილს – *მარცვლის (პიქსელის) ზომა* ეწოდება.

მონიტორების უმნიშვნელოვანესი მახასიათებელია განრჩევალობის უნარი, რომელიც განისაზღვრება მონიტორის სიგრძესა (ჰორიზონტალსა) და სიმაღლეზე (ვერტიკალზე) განთავსებული პიქსელების რაოდენობით თანამედროვე მონიტორების განრჩევალობა **1024 × 768** პიქსელზე ნაკლები არ უნდა იყოს.

■ *თხევადკრისტალურ LCD მონიტორებში* (ნახ. 2.11.გ) გამოიყენება ელექტრული ძაბვის ზეგავლენით სტრუქტურისა და გამჭვირვალობის შეცვლის უნარის მქონე ე.წ. თხევადი კრისტალები. ელექტრული მუხტის არარსებობისას თხევადი კრისტალები ამორფულ მდგომარეობაშია და ატარებს შუქს. მათში გამავალი შუქის რაოდენობა შეგვიძლია ვმართოთ ელექტრული მუხტების გამოყენებით. **LCD** მონიტორის შემნათებელი ნათურები ანათებს ეკრანს, ხოლო თხევადი კრისტალები თითოეულ უჯერდსათვის ატარებს სინათლის მხოლოდ იმ ნაწილს, რომელიც ამ უჯერდისათვის საჭირო სიკაშკაშისთვისაა აუცილებელი. თხევადკრისტალური მონიტორების უმრავლესობა იყენებს მინის ორ ფირფიტას შორის მოთავსებულ თხევადი კრისტალების თხელ აფსკს. გამოსახულების თითოეულ პიქსელს შეესაბამება წითელი, მწვანე და ლურჯი წერტილების გამოხატული სამი უჯერდი.

თხევადკრისტალური მონიტორების უპირატესობაა მათი კომპაქტურობა, დაბალი ენერგომომხარება, ელექტრომაგნიტური გამოსახულების არარსებობა, მაღალი დონის სიკაშკაშე. **CRT-მონიტორებისაგან** განსხვავებით. **LCD-მონიტორებს** აქვს ისეთი მახასიათებელი, როგორცაა **სტანდარტული განრჩევალობა**. მხოლოდ სტანდარტული განრჩევალობის შემთხვევაში აქვს თხევადკრისტალურ მონიტორს გამოსახვის საუკეთესო უნარი. განრჩევალობის შემცირებისას გამოსახულება ბუნდოვანი ხდება. გარდა ამისა, **CRT-მონიტორებთან** შედარებით **CRT-მონიტორები** ფერებს უარესი სიზუსტით გადასცემს. ერთ-ერთ უმნიშვნელოვანესი ნაკლია მზერის კუთხეზე **CRT-მონიტორების** კონტრასტულობის დამოკიდებულება.

■ **აიროპლაზმური GPDF-მონიტორი** თქვენ თუ გადაწყვიტეთ ძველი ტელევიზორი შეცვალოთ უფრო თანამედროვე **საოჯახო კინო-**

**თეატრით**, მაშინ უნდა იფიქროთ მაღალი კლასის გამოქსახველ წაყროზე - **GPDP**-მონიტორზე. იგიშედგება ორი ფირფიტასაგან, რომელთა შორისაც მოთავსებულია ელექტრული იმპულსების ზემოქმედებით მნათი აირების ნარევი. ასეთი მონიტორები თხევადკრისტალური მონიტორებისათვის დამახასიათებელი ნაკლოვანებებსაგან თავისუფალია, მაგრამ მოიხმარს დიდ დენს და აკუმულატორიან და ბატარეებიან გადასატან კომპიუტერებშიმათი გამოყენება არ შეიძლება. ასეთი მონიტორის მაგალითია ფირმა **SAMSUNG**-ის მიერ დამუშავებული **SyncMaster 700IFT** ტიპის მონიტორი. იგი აკმაყოფილებს უსაფრთხოების **TCO 99** სტანდარტს; მისი ელექტრონიკა უზრუნველყოფს **1600 x 1200** პიქსელის ტოლ განრჩევალობას, რაც საკმაოდ კარგი მაჩვენებელია. ასეთი მონიტორების ღირსებებია:

- მაღალი კონტრასტულობა; ამ მხრივ იგი სჯობნის **CRT-LCD** -მონიტორებსა და თანამედროვე ტელევიზორებს; მათი კადრები გამოირჩევა მაღალი ხარისხით, რის გამოც სასურველია ისინი საოჯახო კინოთეატრებში გამოვიყენოთ;

- თავისუფალია თხევადკრისტალური მონიტორებისათვის დამახასიათებელი ინერციულობისაგან;

- თხევადკრისტალური მონიტორებისაგან განსხვავებით: **ა)** მათ აქვს ხედვის ფართო კუთხე; სურათის ხარისხი არაა დამოკიდებული, თუ რომელი კუთხიდან ვუყურებთ მას; **ბ)** მათი გამოსახულება ძალიან მკაფიოა და არ ციმციმებს;

- მათზე ზეგავლენას ვერ ახდენს ელექტრომაგნიტური ველები.

**GPDP**-მონიტორების ნაკლოვანებები აუფერულებს მათ ღირსებებს; კერძოდ, ასეთი მონიტორები ძალიან ძვირია; მოიხმარს დიდ ელექტროენერგიას, რის გამოც მნიშვნელოვნად ხურდება და სჭირდება სპეციალური გამაგრილებელი ვენტილატორები; სშირად იწვება მისი ეკრანის ლუმინოფორი; მათი პიქსელები დიდი ზომისაა, ამიტომ პლანური მონიტორების ზომები **30** ლუიმზე მეტი უნდა იყოს, რის გამოც ოთახი ზომა, რომელშიც ასეთი მონიტორიანი საოჯახო კინოთეატრია მოთავსებული, ძალიან დიდი უნდა იყოს; აქვს ძალიან დიდი წონა.

### პრინტერები

პრინტერი წარმოადგენს ციფრული ინფორმაციის ქაღალდზე გამოქმნა მოწყობილობას. არსებობს მატრიცული, ჭავლური და ლაზერული

პრინტერები.

**მატრიცული პრინტერები** საღებარ ლენტადაფარებულ ქაღალდზე დამრტყმელი მბეჭდავი თავების მეშვეობით გამოსახულების მწკრივებად გამომხატველი მოწყობილობებია. თავი შეიცავს ნემსებს, რომელთა რაოდენობის ზრდის პროპორციულად იზრდება გამოსახულების ხარისხი. უამრავი ნაკლოვანებების (ბეჭდვის დაბალი სიჩქარის, ხმაურის მაღალი დონისა და ა. შ) გამო ამ ტიპის პრინტერები დღეისათვის მოძველდა და გარკვეული სახის ღირსებების მიუხედავად პრაქტიკულად არ გამოიყენება. მათი ღირსებები იყო დაბალი ფასი, პრაქტიკულად ნებისმიერი ხარისხის ქაღალდზე ბეჭდვის უნარი და ა.შ.

**ჭავჭავი პრინტერები** (ნახ. 2.11.დ) მბეჭდავ თავში არსებული მცირე საქმენებიდან მელნის შეფრქვევის გზით ქაღალდზე ბეჭდვის საშუალებაა. შავ-თეთრ პრინტერებში გამოიყენება ერთი (შავი) ფერის, ხოლო ფერად პრინტერებში სამი (ცისფერი, მეწამული და ყვითელი) ფერის მელანი. უფრო ძვირადღირებულ ფერად ჭავჭავ პრინტერებში ზემოთ ჩამოთვლილ სამი ფერის მელანს შავი ფერის მელანიც ემატება, რადგან სუფთად შავი ფერი სამი საბაზისო ფერის შერევით ვერ მიიღება. თავად ჭავჭავი პრინტერები იაფია, ოღონდ ძალიან ძვირია სახარჯო მასალები (მელნით შევსებული კარტრიჯები). გარდა ამისა, ჭავჭავ პრინტერებს სჭირდება მაღალი ხარისხის ქაღალდი, თორემ საღებავი გაიღებება. ჭავჭავი (განსაკუთრებით, ფერადი) პრინტერების ბეჭდვის სიჩქარე ძალიან დაბალია. მიუხედავად აღნიშნული ნაკლოვანებებისა, ჭავჭავი პრინტერები ფართოდაა გავრცელებული, რადგან ისინი უზრუნველყოფს საკმაოდ მაღალი ხარისხის ბეჭდვას.

**ლაზერული პრინტერებში** (ნახ. 2.11.ე) გამოსახულება იქმნება ქაღალდზე სპეციალური ფხვნილის (ტონერის) გადატანის გზით. შუქის წყარო (ლაზერი) ანათებს ფოტოდოლის სპეციალურად დამუხტულ ზედაპირს. იმ ადგილებში, რომლებზეც ეცემა შუქის სხივები, იცვლება მუხტი და მათკენ მიეზიდება ტონერი. იგი ელექტროსტატიკის ხარჯზე გადაიტანება ქაღალდზე; ეს უკანასკნელი შემდეგ ხვდება ლუმელში და მაღალი ტემპერატურის მეშვეობით მასზე მაგრდება ტონერი. ასეთი გამოსახულების ხარისხი ძალიან მაღალია. ლაზერული პრინტერები გამოსახულებას გვერდების და არა



მწკრივების სახით (როგორც ეს ხდება მატრიცულ და ჭავლეურ პრინტერებში) აგებს, რის გამოც მათი მუშაობის სიჩქარე საკმაოდ მაღალია. თანამედროვე პრინტერები წუთში **20-40** გვერდს ბეჭდავენ.

არსებობს მონოქრომული და ფერადი ლაზერული პრინტერები, ოღონდ ეს უკანასკნელები ძალიან ძვირია. სწორედ მაღალი ფასია ლაზერული პრინტერების ძირითადი ნაკლი. მათი ღირსებები, გარდა მაღალი ხარისხით ბეჭდვისა და მაღალი სიჩქარისა, არის ხმაურის დაბალი დონე, მიღებული ანაბეჭდების ხანგამძლეობა, სახარჯი მასალების დაბალი ფასი. ლაზერული პრინტერის კარტრიჯით შეიძლება საკმაოდ დიდი რაოდენობის ანაბეჭდების მიღება.

გრაფიკული ფორმით ქალაქზე, ფირფიტაზე, ფოტომგომნობიარე მასალაზე ან სხვა მზიდზე მელნის საშუალებით გრაფიკული ფორმით მონაცემები გამოიტანება *პლოტერის* (ნახ. **2.11.ვ**) საშუალებით.

## ლიტერატურა

1. **დუნდუა ა.** კომპიუტერული სისტემებისა და საინფორმაციო ტექნოლოგიების თეორიული საფუძვლები – თბ. „ტექნიკური უნივერსიტეტი“, **2014.** – 258 გვ.
2. **დუნდუა ა.** ტრანსპორტზე მიკროპროცესორული ტექნიკის გამოყენების საფუძვლები - თბ. „ტექნიკური უნივერსიტეტი“, **2017.** – 344 გვ.
3. **მაჭარაძე თ., წვერაიძე ზ.** ინფორმატიკის საფუძვლები. - თბ.„ტექნიკური უნივერსიტეტი“, **2009.** – 342 გვ.
4. **მაჭარაძე თ., წვერაიძე ზ.** კომპიუტერები და კომპიუტერული ტექნოლოგიები. - თბ.„ტექნიკური უნივერსიტეტი“, **2009.** – 363 გვ.
5. **Грошев А.С., Закляков П.В.** Информатика. – М. ДМКПресс, **2014.** – 592 с.
6. Информатика / под ред. **Трофимова В.В.** – М. Издательство Юрайт, **2013.** – 917 с.
7. **Ляхович В.Ф., Молодцов В.А., Рыжкова Н.Б.** Основы информатики. – М. КНОРУС, **2016** – 348 с.
8. **Новожилов О.П.** Архитектура ЭВМ и систем.– М. Издательство Юрайт, **2016.** – 537 с.
9. **Новожилов О.П.** Информатика.– М.: Издательство Юр-айт, **2012.** – 564 с.
10. **Таненбаум Э., Остин Т.** Архитектура компьютера. – СПб. Питер, **2013.** – 816 с.
11. **Хлебников А.А.** Информационные технологии. - М. КНОРУС, **2014.** – 472 с.

რედაქტორი ბ. ცხადაძე

კომპიუტერული უზრუნველყოფა ალექსანდრე ღუნდუასი

გადაეცა წარმოებას 05.08.2019. ხელმოწერილია დასაბეჭდად  
03.10.2019. ქალაქის ზომა 60X84 1/16. პირობითი ნაბეჭდი თაბახი 9.  
№3148.

საგამომცემლო სახლი „ტექნიკური უნივერსიტეტი“,  
თბილისი, კოსტავას 77



Verba volant.  
scripta manent