

**თემა 2. ციფრული ელექტრონიკის ელემენტები - ინფორმაციის გარდაქმნის და დამუშავების ზოგიერთი უმარტივესი ხელსაწყო**

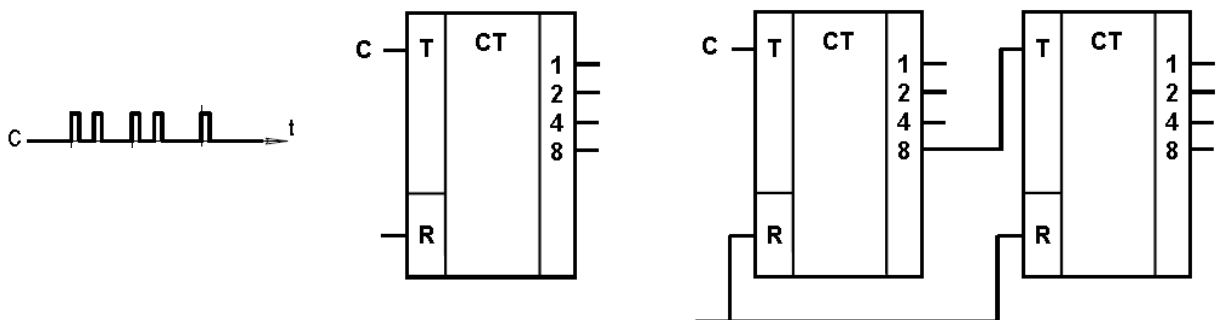
**მეცადინეობა 2.**

ციფრული ინფორმაციის გარდაქმნის, დამუშავების და ურთიერთგაცვლისათვის შექმნილია სხვადასხვა ტიპური ელექტრული ხელსაწყო. მათი წყება წარმოადგენს თავისებურ “კონსტრუქტორს” – მათი ურთიერთშეერთებით ახალი ფუნქციონალური თვისებების მქონე ელექტრული კონსტრუქციების მიღებაა შესაძლებელი. აუცილებლად მკაცრად უნდა იყოს დაცული ამ საწყისი, ელემენტარული ბლოკების სხვებთან მიერთების და ფუნქციონირების წესი. წინა მეცადინეობაზე გავიცანით შიფრატორები და დეშიფრატორები, და მათი გამოყენებით ავაწყეთ შედარებით რთული ფუნქციონალური კვანძი. ქვემოთ გავეცნობით კიდევ რამდენიმე ხელსაწყოს და კიდევ ავაწყობთ სხვადასხვა დანიშნულების კვანძებს.

**მთვლელო**

ეს ელექტრონული ხელსაწყო იძლევა საშუალებას დაეთვალოთ და ორობითი რიცხვის სახით გამოვხატოთ მის შესავალზე თანმიმდევრულად მიწოდებული ელექტრონული იმპულსების რაოდენობა. იმპულსების თანმიმდევრობა და მთვლელის გრაფიკული გამოსახულება წარმოდგენილია ნახ. 2.10-ზე.

ნახ. 2.10



1. იმპულსების თანმიმდევრობა                      2. მთვლელო                      3. მთვლელის მიმდევრობითი ჩართვა

თუ მთვლელის შესავალზე (ელექტროდი **C**) მიწოდებული სიგნალი შეიცავს 5 იმპულსს, მთვლელის გამოსავალზე მიღებული იქნება ორობითი რიცხვი **101**, რომელიც იმპულსების რაოდენობის ტოლია. ნახატზე გამოსახული მთვლელო, იმპულსების რაოდენობის გაზრდის შემთხვევაში, დაითვლის მათ რაოდენობას 16-მდე, და ისევ თავიდან დაიწყებს თვლას. მთვლელო რეაგირებს შესავალი იმპულსის არა დასაწყისზე (ფრონტზე), არამედ მის დასასრულზე – ძაბვის დაცემაზე. მთვლელს გააჩნიათ მართვის შესავალი **R**, რომლის საშუალებით შესაძლებელია მთვლელის საწყის მდგომარეობაში გადაყვანა – დათვლილი იმპულსების რიცხვის წაშლა. ამ შესავალზე იმპულსის მიწოდების შემდეგ გამოსავალ ელექტროდებზე მივიღებთ ნოლებს.

შესაძლებელია მთვლელის მიმდევრობითი ჩართვა. ამ შემთხვევაში ყოველ შემდგომ მთვლელს მიეწოდება ინფორმაცია წინა მთვლელის უფროსი თანრიგიდან, ესე იგი წინა მთვლელის თვლის მაქსიმალური შედეგი ერთი

იმპულსის სახით გადაეცემა მეორე მთვლელს. თვლის დროს, რიგით მეოთხე ელექტროდზე, გაჩნდება 1 (გაიზრდება ძაბვა) როდესაც დაითვლება 8 იმპულსი. 1 შეიცვლება 0-ით როდესაც შესავალზე დამთავრდება მეოქსემეტე იმპულსი. სწორედ ამ მომენტში მეორე მთვლელის გამოსავალზე, მის პირველ გამოსავალ ელექტროდზე, გაჩნდება 1. ორ თანმიმდევრულად ჩართულ მთვლელს შეუძლია დათვალოს  $16 \cdot 16 = 256$  იმპულსი. სამ მთვლელს შეუძლია  $16 \cdot 16 \cdot 16 = 4096$  იმპულსის დათვლა. ასეთი ჩართვის შემთხვევაში ერთმანეთთან აერთიანებენ მართვის ელექტროდებს, რათა ასეთი კომპლექსური მთვლელის საწყის მდგომარეობაში მოყვანა ხდებოდეს სინქრონულად.

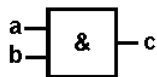
არსებობენ მთვლელები, რომლებიც ითვლიან ათამდე და უბრუნდებიან საწყის ნულოვან მდგომარეობას. სამ ასეთ მთვლელს შეუძლია დათვალოს 1000 იმპულსი.

შესაძლებელია მთვლელის თვლის კოეფიციენტის შეცვლა დამატებითი ლოგიკური ელემენტების გამოყენებით.

ერთერთ ასეთ ლოგიკურ ელემენტს წარმოადგენს ლოგიკური ელემენტი “და”. გამოყენების თვალსაზრისით უფრო ტექნოლოგიურია ლოგიკური ელემენტი “და-არა”. ეს ლოგიკური ელემენტები წარმოდგენილია ნახ. 2.11 –ზე. მათი ფუნქციონირების ცხრილები და ლოგიკური ოპერაციის აღწერა იქვეა წარმოდგენილი.

ნახ. 2.11

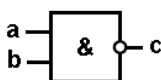
1. ელემენტი “და”



a	b	C
0	0	0
0	1	0
1	0	0
1	1	1

ლოგიკური ელემენტი “და” ადარებს შემავალ სიგნალებს და რეაგირებს იმ შემთხვევაში, როდესაც შესავალის ორივე ელექტროდზე არის სიგნალი, ე.ი. მიწოდებულია 1.

2. ელემენტი “და-არა”



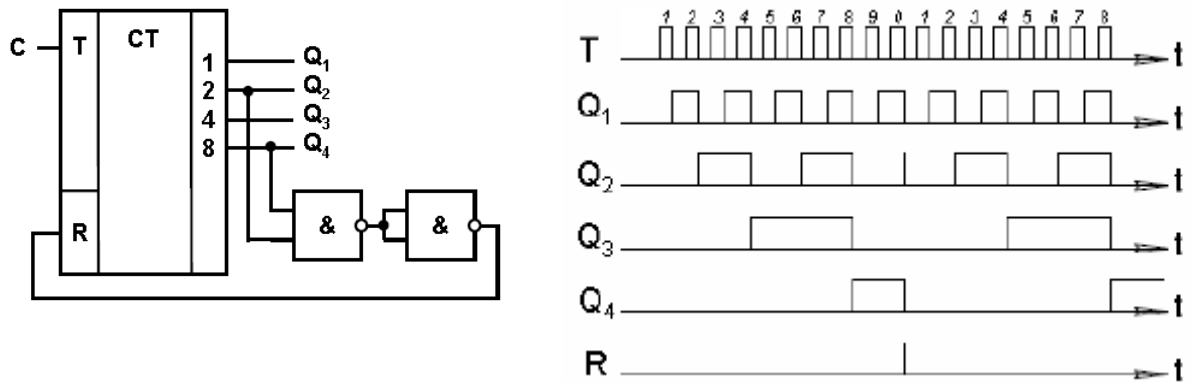
a	b	C
0	0	1
0	1	1
1	0	1
1	1	0

ლოგიკური ელემენტი “და-არა” ასრულებს ელემენტ და-ს შებრუნებულ ოპერაციას (უარყოფს და-ს). ადარებს რა შემავალ სიგნალებს, რეაგირებს 1-ანით გამოსავალზე იმ შემთხვევაში, როდესაც შესავალის ორივე ელექტროდზე ან არ არის სიგნალი, ან სიგნალი არის მხოლოდ ერთზე. თუ ორივე შესავალზე 1-ა, გამოსავალზე იქნება 0.

ამ ელემენტების გამოყენებით შესაძლებელია მთვლელების “გადაპროგრამება”. მაგალითად, გადავაპროგრამოთ მთვლელი, რომელიც ითვლის თექვსმეტამდე ისე, რომ ათამდე თვალოს. ეს იმას ნიშნავს, რომ ათამდე დათვლის შემდეგ მთვლელის გამოსავალ ელექტროდებზე ნოლები უნდა გაჩნდეს – მთვლელი საწყის მდგომარეობას უნდა დაუბრუნდეს. ამის მიღწევა შესაძლებელია **R** შესავალი ელექტროდის გამოყენებით. ნახ. 2.12-ზე. ნაჩვენებია მთვლელის თვლის კოეფიციენტის გადაპროგრამების მიღწევის ხერხი და ნაჩვენებია თვლის დიაგრამა, ანუ მთვლელის გამოსავალი ელექტროდების მდგომარეობის ცვლილება დროში. ლოგიკური ელემენტი მიერთებულია ორ შერჩეულ გამოსავალ ელექტროდთან, რომლებიც შეესაბამებიან ორობითი რიცხვის მეორე

და მეოთხე თანრიგებს. თვლის დროს, როგორც კი მეათე შესავალი იმპულსი მიეწოდება მთვლელს, მის გამოსავალზე გაჩნდება ორობითი რიცხვი 1010. ლოგიკური ელემენტი მაშინათვე მოახდენს რეაგირებას და გადაიყვანს მთვლელს საწყის ნულოვან მდგომარეობაში. თვლის შემდეგი ციკლი ისევ ათამდე გაგრძელდება. გაითვალისწინეთ, რომ მთვლელი, როგორც “შავი ყუთი”, მუშაობისას ყოველთვის აჯამებს შესული იმპულსების რაოდენობას და გამოყავს ამ ჯამის მნიშვნელობა ორობითი რიცხვის სახით. მიაქციეთ ყურადღება მთვლელის მუშაობის დიაგრამას:

ნახ. 2.12

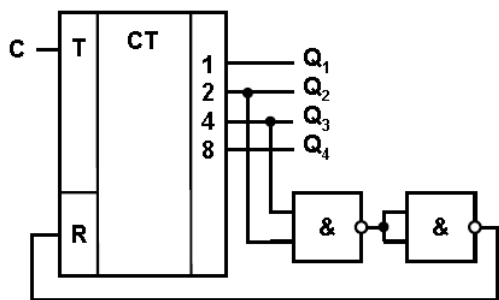


- შესავალზე პირველი მოსული იმპულსის დამთავრებისთანავე, პირველ გამოსავალ ელემენტოდზე  $Q_1$  –ზე ჩნდება 1. დანარჩენ გამოსავალ ელემენტოდებზე გვაქვს ნოლები. პირველი იმპულსი დათვლილია.
- $Q_1$  ელემენტოდის მდგომარეობა შეიცვლება 0 –ით შესავალზე მოსული მეორე იმპულსის დამთავრებისთანავე. ამავდროულად 1 გაჩნდება მეორე ელემენტოდზე –  $Q_2$  –ზე, და მთვლელის გამოსავალზე გვექნება ორობითი რიცხვი 10.
- შესავალზე მესამე იმპულსის დამთავრებისთანავე 1 ისევ ჩნდება პირველ ელემენტოდზე. რადგან ელემენტოდი  $Q_2$  ინარჩუნებს თავის მდგომარეობას, ამ მომენტიდან მთვლელის გამოსავალზე გვაქვს ორობითი რიცხვი 11.
- შესავალზე მეოთხე იმპულსის დამთავრებისთანავე  $Q_1$  და  $Q_2$  გადადიან ნულოვან მდგომარეობაში და 1 ჩნდება  $Q_3$  ელემენტოდზე. მიღებულია ორობითი რიცხვი 100.
- ამ წესით თვლა გრძელდება. თანმიმდევრულად მიიღება რიცხვები  $101_2=5$ ,  $110_2=6$ ,  $111_2=7$ .
- შესავალზე მერვე იმპულსის დამთავრებისთანავე 1 გაჩნდება  $Q_4$  ელემენტოდზე და მივიღებთ ორობით რიცხვს 1000.
- შესავალზე მეცხრე იმპულსის დამთავრებისთანავე 1 გაჩნდება  $Q_1$  ელემენტოდზეც და მივიღებთ ორობით რიცხვს 1001.
- შესავალზე მეათე იმპულსის დამთავრებისთანავე და  $Q_2$  ელემენტოდზე 1-ის გაჩენით მაშინადვე ამუშავდება ლოგიკური ელემენტი და მთვლელი გადავა საწყის მდგომარეობაში

ასეთი მიდგომის გამოყენებით შესაძლებელია თვლის სხვა კოეფიციენტის დაპროგრამება. მაგალითად 6-მდე თვლის სქემა მოყვანილია ნახ. 2.13 –ზე. თვლა იგივე ალგორითმით გრძელდება. ლოგიკური ელემენტი ამუშავდება და გადაიყვანს მთვლელს საწყის ნულოვან მდგომარეობაში მეექვსე იმპულსის დამთავრებისთანავე.

მოვლელთან დეშიფრატორის და რიცხვითი ინდიკატორის მიერთება არავითარ სიძნელეს არ წარმოადგენს. სქემის ასეთი გარდაქმნის შემთხვევაში ჩვენ მივიღებთ მოვლელების ფართო დანიშნულების ოჯახს, რომელიც გამოიყენება ტექნიკის სხვადასხვა დარგში. თუ, მაგალითად შესასვლელი აღჭურვილია მბრუნავი მოწყობილობით, რომელიც ერთი ადამიანის გასვლის დროს ერთხელ მობრუნდება და მოგვცემს სასიგნალო იმპულსს, შეგვიძლია ვთვალოთ რამდენმა ადამიანმა გაიარა შესასვლელი. თუ მოვლელს მიეწოდება იმპულსები 1 სეკუნდის ინტერვალით, შეგვიძლია ავაწყოთ საათი. მაგალითი მოყვანილია ნახ. 2.14 –ზე.

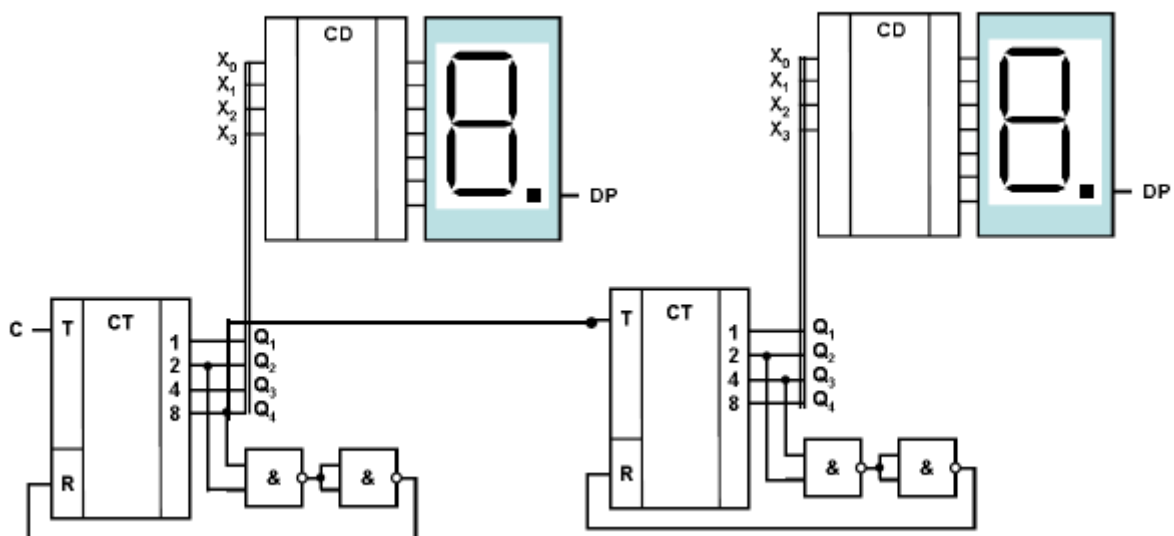
ნახ. 2.13



1. შესავალზე პირველი მოსული იმპულსის დამთავრებისთანავე, პირველ გამოსავალ ელექტროდზე  $Q_1$ -ზე ჩნდება 1. დანარჩენ გამოსავალ ელექტროდებზე გვაქვს ნოლები. პირველი იმპულსი დათვლილია.

2.  $Q_1$  ელექტროდის მდგომარეობა შეიცვლება 0 – ით შესავალზე მოსული მეორე იმპულსის დამთავრებისთანავე. ამავედროულად 1 გაჩნდება მეორე ელექტროდზე –  $Q_2$  –ზე, და მოვლელის გამოსავალზე გვექნება ორობითი რიცხვი 10.
3. შესავალზე მესამე იმპულსის დამთავრებისთანავე 1 ისევ ჩნდება პირველ ელექტროდზე. რადგან ელექტროდი  $Q_2$  ინარჩუნებს თავის მდგომარეობას, ამ მომენტიდან მოვლელის გამოსავალზე გვაქვს ორობითი რიცხვი 11.
4. შესავალზე მეოთხე იმპულსის დამთავრებისთანავე  $Q_1$  და  $Q_2$  გადადიან ნულოვან მდგომარეობაში და 1 ჩნდება  $Q_3$  ელექტროდზე. მიღებულია ორობითი რიცხვი 100.
5. შემდეგ მიიღება რიცხვი  $101_2=5$ .
6. შესავალზე მეექვსე იმპულსის დამთავრებისთანავე და  $Q_2$  ელექტროდზე 1-ის გაჩენით აშუშავდება ლოგიკური ელემენტი და მოვლელი გადავა საწყის მდგომარეობაში

ნახ. 2.14



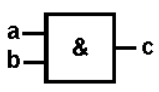
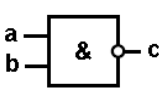
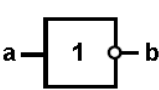
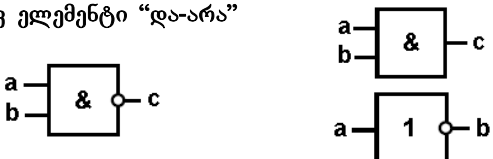
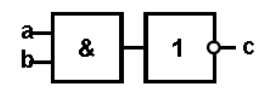
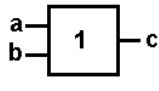
წრედის პირველი რგოლი ითვლის ათამდე, მეორე კი ექვსამდე. თუ ამ სქემის შესავალზე მივაწვდით იმპულსებს ყოველ წამს, მივიღრებთ წამმზომს. თუ გავზრდით რგოლების

რაოდენობას და შევარჩევთ თვლის სათანადო კოეფიციენტებს – ავაწყობთ საათს. რა თქმა უნდა საათის ციფერბლატზე პირველ ადგილას უნდა განთავსდეს საათების თვლის შედეგი – მთვლელის ბოლო რგოლის ინდიკატორი. ის მხოლოდ ორზე გაყოფას უნდა ასრულებდეს (დღეღამე 24 საათია). მარცხნიდან მეორე პოზიციას დაიკავენ ინდიკატორი, რომლის მთვლელი ითვლის ათამდე. შემდეგი რგოლი დათვლის 6-მდე (საათში 60 წუთია), და ასე შემდეგ. ასეთი ციფერბლატის მაქსიმალური რიცხვია 23 საათი 59 წუთი 59 წამი.

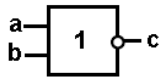
მიაქციეთ ყურადღება, რომ ლოგიკური წრედი შედგენილია არა ართი “და” ელემენტისაგან, არამედ ორი “და-არა” ელემენტისაგან. პირველი შეხედვით, სქემა ამით გართულებულია, მაგრამ ტექნოლოგიურად უფრო მოხერხებული აღმოჩნდა მიკროელექტრონული სქემების (ჩიპების) მოხმარება, რომლებშიც რამდენიმე “და-არა” ელემენტი მოთავსებული. ელემენტების კომბინაციით ორივე ლოგიკური ელემენტი მიიღება. თუ ჩიპში მხოლოდ “და” ელემენტებია, იგივე ჩიპით “და-არა” ელემენტებს ვერ მივიღებთ.

ელექტრონიკაში სხვა მარტივი ლოგიკური ელემენტებიც გამოიყენება. მათ შესავალ ელექტროდებზე ინფორმაცია ერთდროულად მიეწოდება, ხოლო შედეგი გამისავალზე, როგორც “შავი ყუთის” გამოსავალზე, შეესაბამება ამა თუ იმ ლოგიკური ოპერაციის განმარტებას. ნახ. 2.15-ზე გამეორებულია ნახ. 2.11 ელემენტები და წარმოდგენილია რამდენიმე ახალი.

ნახ. 2.15

<p>1. ელემენტი “და”</p> 	<table border="1"> <thead> <tr> <th>a</th> <th>b</th> <th>c</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	a	b	c	0	0	0	0	1	0	1	0	0	1	1	1	<p>ლოგიკური ელემენტი “და” ადარებს შემავალ სიგნალებს და რეაგირებს იმ შემთხვევაში, როდესაც შესავალის ორივე ელექტროდზე არის სიგნალი, ე.ი. მიწოდებულია 1.</p>
a	b	c															
0	0	0															
0	1	0															
1	0	0															
1	1	1															
<p>2. ელემენტი “და-არა”</p> 	<table border="1"> <thead> <tr> <th>a</th> <th>b</th> <th>c</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	a	b	c	0	0	1	0	1	1	1	0	1	1	1	0	<p>ლოგიკური ელემენტი “და-არა” ასრულებს ელემენტ და-ს შებრუნებულ ოპერაციას (უარყოფს და-ს). ადარებს რა შემავალ სიგნალებს, რეაგირებს 1-ნით გამოსავალზე იმ შემთხვევაში, როდესაც შესავალის ორივე ელექტროდზე ან არ არის სიგნალი, ან ერთზე არის სიგნალი. თუ ორივე შესავალზეა 1, გამოსავალზე 0.</p>
a	b	c															
0	0	1															
0	1	1															
1	0	1															
1	1	0															
<p>3. ელემენტი “არა”</p> 	<table border="1"> <thead> <tr> <th>a</th> <th>B</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> </tr> </tbody> </table>	a	B	0	1	1	0	<p>უმარტივესი ლოგიკური ელემენტი.</p>									
a	B																
0	1																
1	0																
<p>4. ისევ ელემენტი “და-არა”</p> 		<p>ეს ელემენტი წარმოადგენს ორი ელემენტის კომბინაციას – და, არა.</p>															
<p>5. ელემენტი “ან”</p> 	<table border="1"> <thead> <tr> <th>a</th> <th>b</th> <th>c</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	a	b	c	0	0	0	0	1	1	1	0	1	1	1	1	<p>ელემენტი “ან” რეაგირებს იმ შემთხვევაში, თუ ინფორმაცია შესავალ ელექტროდებზე განსხვავდება.</p>
a	b	c															
0	0	0															
0	1	1															
1	0	1															
1	1	1															

6. ელემენტი “ან-არა”



a	b	c
0	0	1
0	1	0
1	0	0
1	1	0

ელემენტი “ან-არა” წარმოადგენს ან-ის უარყოფას. რეაგირებს იმ შემთხვევაში, თუ ინფორმაციი შესავალ ელექტროდებზე არ განსხვავდება, ან არ არის, ე.ი. ორივე ელექტროდზე 0. ეს ელემენტიც წარმოადგენს ორი ელემენტის კომბინაციას. “ან” + “არა”.

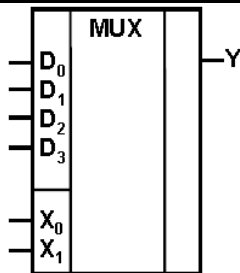
ჩამოთვლილ ლოგიკურ ელემენტებს შესაძლებელია გააჩნდეს არა ორი, არამედ მეტი შესავალი ელექტროდი. ამ შემთხვევებში სათანადო ცხრილებს უფრო რთული სახე ექნება.

შესწავლილ ელექტრონულ მოწყობილობებს ემატება ორი ახალი.

**მულტიპლექსორი**

ამ ხელსაწყოს (ნახ. 2.16) დანიშნულებაა პოზიციური კოდის სახით წარმოდგენილი ინფორმაციის გარდაქმნა იმპულსების მიმდევრობად. სიტყვები multi და plexor სათანადოთ ნიშნავენ მრავალს და პატარა ჩაქუნს, რომელიც გამოიყენება მედიცინაში (plexis – ბერძნ. დარტყმა). ხელსაწყოს გააჩნია ორი შესავალი ბლოკი – პორტი. პირველი ინფორმაციულია, და მას პოზიციური კოდის სახით მიეწოდება შესავალი ინფორმაცია. მეორე პორტით იმართება მულტიპლექსორის მუშაობა. ამ პორტზე მიწოდებული კოდის შესაბამისად ინფორმაცია შესავლის სათანადო ელექტროდიდან გადაიცემა გამოსავალზე. მაგალითად, თუ მართვის კოდი წარმოდგენილია რიცხვებით 00 – გამოსავალზე გამოყვანილი გვექნება  $D_0$  ელექტროდზე არსებული ინფორმაცია. თუ მართვის კოდი წარმოდგენილია ორობითი რიცხვებით 01 ( $X_0$ -ს შეესაბამება 1, ხოლო  $X_1$ -ს შეესაბამება 0) – გამოსავალზე გამოყვანილი გვექნება  $D_1$  ელექტროდზე არსებული ინფორმაცია, და ასე შემდეგ. გასაგები ხდება ხელსაწყოს დასახელების არსი – მართვის შესავალზე “დაკაკუნებით” თანმიმდევრულად “გამოიყვანება” შესავალზე არსებული ინფორმაცია.

ნახ. 2.16



$X_1$	$X_0$	Y
0	0	$D_0$
0	1	$D_1$
1	0	$D_2$
1	1	$D_3$

შესაძლებელია მულტიპლექსორის ელექტროდების რაოდენობის გაზრდა, მაგალითად 16 საინფორმაციო ელექტროდი და, სათანადოდ 4 მართვის ელექტროდი (რომლებიც ორობითი წარმოდგენით 16 ელექტროდის მართვის საშუალებას იძლევა).

მაიქციეთ ყურადღება, რომ მილტიპლექსორი, ინფორმაციის ზუსტად გადასაცემად, უნდა ასრულებდეს ციკლურ მუშაობას შემდეგი წესით:

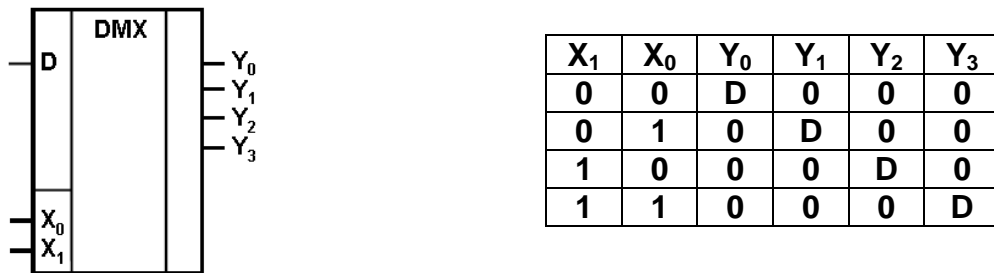
- ინფორმაცია საინფორმაციო ელექტროდებზე არ უნდა იცვლებოდეს “გამოყვანის” პროცედურის განმავლობაში.

- პირველი ეტაპი - ციკლის დასაწყისში უნფორმაცია შემოდის საინფორმაციო ელექტროდებზე და უცვლელი რჩება ციკლის დამთავრებამდე.
- მეორე ეტაპი – რაიმე საშუალებით იცვლება ორობითი რიცხვი მართვის ელექტროდებზე და ამ რიცხვის შესაბამისი საინფორმაციო ელექტროდიდან ინფორმაცია გამოდის მულტიპლექსორის გამოსავალზე.
- ციკლის დასრულებისას შესაძლებელია ახალი ციკლის დაწყება – ახალი ინფორმაციის მიწოდება საინფორმაციო ელექტროდებზე და ინფორმაციის გამოყვანის დაწყება.

### დემულტიპლექსორი

ეს მოწყობილობა მულტიპლექსორის მსგავსია, მაგრამ მისი ინფორმაციის გარდაქმნის ალგორითმი შებრუნებულია – ინფორმაცია, რომელიც წარმოდგენილია იმპულსების მიმდევრობის სახით, გარდაიქმნება და წარმოსდგება პოზიციური კოდის სახით (ნახ. 2.17).

ნახ. 2.17



მიაქციეთ ყურადღება, რომ დემულტიპლექსორის გამოსავალზე ინფორმაცია არ არის ერთდროულად წარმოდგენილი ორობითი რიცხვით და ციკლურად იცვლება. ციკლს განსაზღვრავს მართვის ელექტროდების მდგომარეობა:

- ვაწვდით მართვაზე ორობით 00, შესავალზე ინფორმაციას და ვკითხულობთ ამ ინფორმაციას გამოსავლის პირველ ელექტროდზე.
- ვაწვდით მართვაზე ორობით 01, შესავალზე ახალ ინფორმაციას და ვკითხულობთ ამ ინფორმაციას გამოსავლის მეორე ელექტროდზე.
- ვაწვდით მართვაზე ორობით 10, შესავალზე ახალ ინფორმაციას და ვკითხულობთ ამ ინფორმაციას გამოსავლის მესამე ელექტროდზე.
- ვაწვდით მართვაზე ორობით 11, შესავალზე ახალ ინფორმაციას და ვკითხულობთ ამ ინფორმაციას გამოსავლის მეოთხე ელექტროდზე.

ადვილი მისახვედრია, რომ ამ ორი მოწყობილობის გამოყენებით შესაძლებელი ხდება უმარტივესი ინფორმაცია გადაცემის არხის ორგანიზაცია. თუ მულტიპლექსორი და დემულტიპლექსორი ერთმანეთს დაშორებულია, მაგრამ მათი მართვა რაღაცნაირად სინქრონიზირებულია, ანუ ერთდროულად ერთი და იგივე მართვის კოდი მიეწოდება, მაშინ არხის შესავალზე არსებული ინფორმაცია გადაიცემა და აღდგება არხის გამოსავალზე. ამის სქემატური, გამარტივებული ილუსტრაცია წარმოდგენილია ნახ. 2.18 –ზე. ქვემოთ მსგავსი საინფორმაციო არხის ორგანიზაციას განვიხილავთ დეტალურად. ასეთი კავშირი და არხების ორგანიზაცია ხშირად გამოიყენება ელექტრონულ ტექნიკაში.

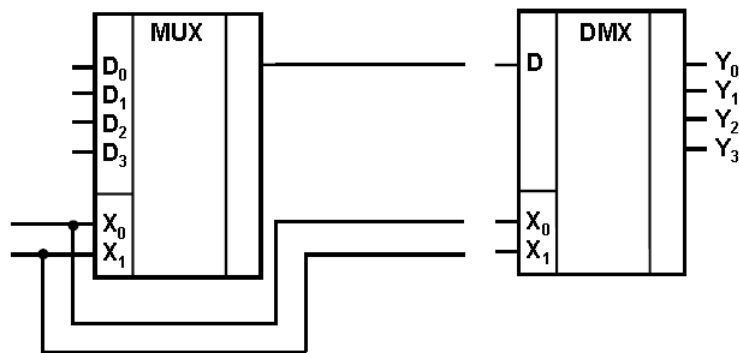
არხის სინქრონიზაცია შესაძლებელია განხორციელდეს მთვლელების გამოყენებით. ამ შემთხვევაში საჭიროა დავაყენოთ ერთი მთვლეელი

მულტიპლექსორის სამართავად და მეორე დემულტიპლექსორის სამართავად. ირივე მთვლელს უნდა მიეწოდოს ერთი და იგივე წყაროდან იმპულსები და მათი საწყის მდგომარეობაში დასაყენებელი ელექტროდები უნდა გაერთიანდეს.

### დავალება 1

დახაზეთ მთვლელებით სინქრონიზირებული არქის სქემა და აღწერეთ მისი მუშაობა.

ნახ. 2.18



ლოგიკური ელემენტების გამოყენებით უკვე შესაძლებელია პირველი, შედარებით რთული სქემების აგება. მაგალითისათვის განვიხილოთ მულტიპლექსორის სქემა. სქემა მოყვანილია ნახ. 2.19 –ზე. მულტიპლექსორის ასაგებად გამოყენებულია:

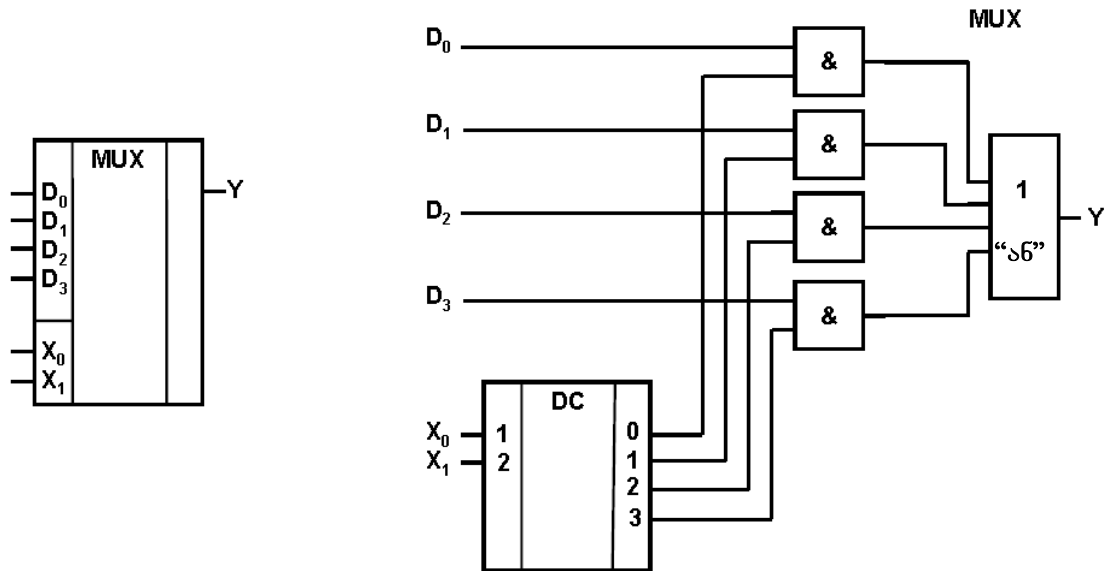
- ერთი დეშიფრატორი, რომელსაც მართვის ორობითი კოდი გადაყავს ათობითში;
- ოთხი ელემენტი “და”, რომელთა მეშვეობით ხდება შესავალი ინფორმაციის “წაკითხვა”;
- ერთი ელემენტი “ან”, რომელსაც საბოლოოდ გამოყავს ინფორმაცია მულტიპლექსორიდან.

სქემა ფუნქციონირებს შემდეგნაირად:

- დეშიფრატორი, შესავალი კოდის შესაბამისად, თანმიმდევრულად აწარმოებს ლოგიკური ელემენტების “გამოკითხვას” – შესავალი კოდის შესაბამისად სათანადო ელემენტის შესატყვის ელექტროდზე მიეწოდება 1, დანარჩენ ლოგიკურ ელემენტებს მიეწოდებათ 0;
- ის ლოგიკური ელემენტი, რომელიც გააქტიურებულია (რომელსაც მიეწოდა 1), იწყებს ფუნქციონირებას – თუ მის მეორე ელექტროდზე არსებული ინფორმაცია აგრეთვე წარმოდგენილია 1-ით, “და” ელემენტის გამოსავალზე გაჩნდება 1;
- შემდეგ ეტაპზე სიგნალი შედის “ან” ელემენტში და გამოდის მულტიპლექსორის გამოსავალზე, ეს ელემენტი მხოლოდ აერთიანებს “და” ელექტროდების გამოსავლებს ინფორმაციის მულტიპლექსორიდან გამოსაყვანად

ნახ. 2.19

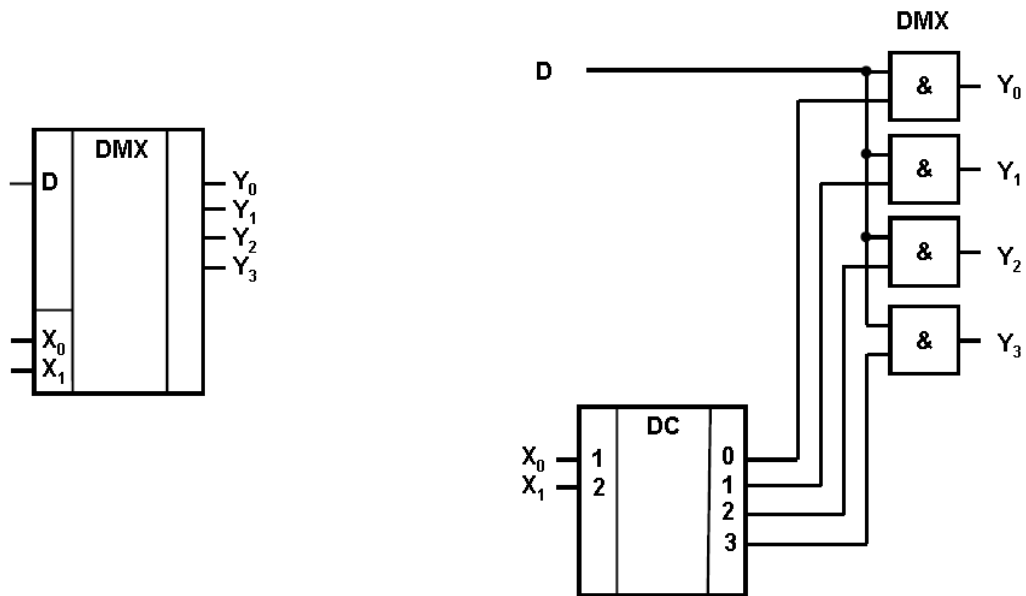




**დავალება 2**

აღწერეთ ნახ. 2.20-ზე წარმოდგენილი დემულტიპლექსორის მუშაობის თანმიმდევრობა.

ნახ. 2.20

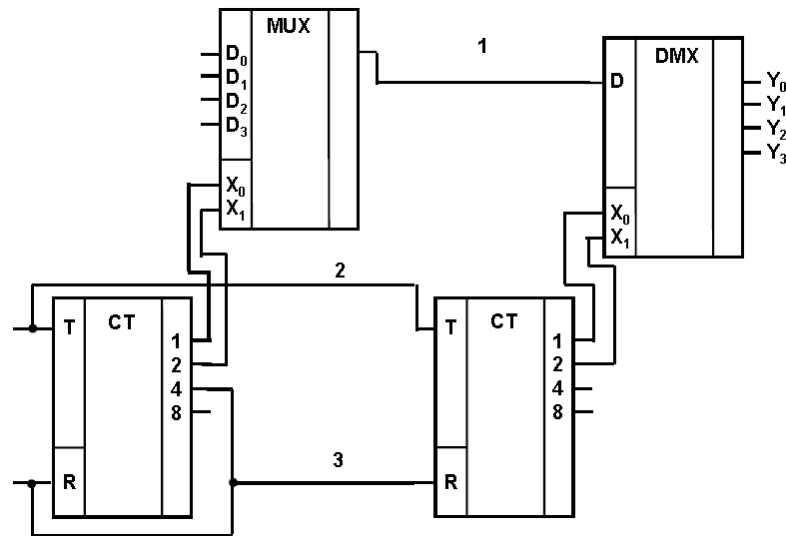


**დავალების პასუხი 1**

გადამცემი არხის სინქრონიზაცია მოვლელების გამოყენებით.

მოვლელებით სინქრონიზაციისათვის საჭიროა არხის ორივე ბოლოს განთავსდეს მოვლელი, რომლის გამოსავალი კოდი გამოყენებული იქნება მულტიპლექსორის და დემულტიპლექსორის სამართავად. მოვლელების გამოსავალი ელექტროდების

პოზიციები შეთანხმებული უნდა იყოს სამართავი მოწყობილობების პოზიციებთან. ამ შემთხვევაში მთვლელებზე მიწოდებული იმპულსების რაოდენობის შესაბამისად იმუშავენ არხის მართვა. რადგან მართვა ხორციელდება მხოლოდ ორი ორობითი თანრიგით, მთვლელების თვლის კოეფიციენტი არ უნდა აღემატებოდეს 3. ეს უზრუნველყოფილია მთვლელების საწყის მდგომარეობაში გადაყვანის ელექტროდების შეერთებით და ამ წრედის შეერთებით მთვლელის გამოსავალ მესამე ელექტროდთან, რომლის ათობითი მნიშვნელობა უდრის 4. აუცილებელია მთვლელების შესავლების გაერთიანება. შექმნილ სქემაში 1 არხით გადაიცემა ინფორმაცია, 2 არხით თვლადი იმპულსები, და 3 არხით განხორციელდება მთვლელების სინქრონიზაცია.



## დავალების პასუხი 2

ნახაზ 2.20-ზე მოყვანილი დემულტიპლექსორის ლოგიკური ელემენტები იმართებიან დეშიფრატორით. მართვის კოდის შესაბამისად დეშიფრატორი თანმიმდევრულად აწვდის 1 ლოგიკური ელემენტების შესავლებზე. რადგან ლოგიკური ელემენტების მეორე შესავლები გართიანებულია და წარმოადგენს დემულტიპლექსორის საინფორმაციო შესავალს, გააქტიურებული – ინფორმაციის გამოსაყვანად მომზადებული, ყოველთვის იქნება ის ელემენტი რომელსაც დეშიფრატორი მიმართავს. თუ გააქტიურების მომენტში ინფორმაცია წარმოდგენილია 1-ით, ის გავა გამოსავალზე, თუ 0-ით, ელემენტის გამოსავალზე დარჩება 0.