<http://www.electronics-tutorials.ws>

დამატებულია მასალა წიგნიდან: Ревич Ю.[Занимательная электроника](http://www.books.ru/shop/books/239906), 2005

ბულის ალგებრა **Tutorial: 1 of 7**

**შესავალი**

[](http://upload.wikimedia.org/wikipedia/commons/6/6c/George_Boole.jpg)ჯორჯ ბული[](http://upload.wikimedia.org/wikipedia/en/2/2f/Claude_Elwood_Shannon_(1916-2001).jpg)კლოდ შენონი

ადამიანებს ყოველთვის აწუხებდა აზროვნების საიდუმლოება და პირველ რიგში იმ კანონზომიერებების დადგენა და მათემატიკური სახით ჩამოყალიბება, რომლებიც განაპირობებენ და მართავენ აზროვნების პროცესს. საბოლოო ჯამში, ყველაფერი ეს ჩამოყალიბდა მეცხრამეტე საუკუნის ინგლისელი მათემატიკოსის ჯორჯ ბულის ორ შრომაში. ჩამოყალიბებული მათემატიკური ფორმალიზმი იგივდებოდა აზროვნების კანონზომიერებებთან. ეს წარმოდგენა პირდაპირ აისახა შრომის სახელწოდებაში: [*An Investigation of the Laws of Thought on Which are Founded the Mathematical Theories of Logic and Probabilities*](http://www.gutenberg.org/etext/15114)*.* უნდა აღინიშნოს, რომ მხოლოდ მეოცე საუკუნის პირველი ნახევრის მიწურულს გამოვლინდა, რომ აზროვნებას არ გააჩნია ლოგიკური ბუნება, და ლოგიკა მხოლოდ აზრის გადმოცემის ხელსაყრელი ხერხია, რომელიც გვაძლევს საშუალებას სიტყვიერი ფორმით ცალსახად ჩამოვაყალიბოდ და გადავცეთ ინფორმაცია.

თითქმის ერთი საუკუნის განმავლობაში ვერ მოხდა იმის გააზრება, რომ ბულის მიერ განვითარებული მათემატიკური ლოგიკის ფორმალიზმი ექვემდებარება ტექნიკურ რეალიზაციას და სასიგნალო არხების კომუტაციის ტექნიკური ამოცანების გადასაჭრელად შეიძლება იყოს გამოყენებული. ბულის ალგებრის კანონები ზუსტად ასახავენ კომუტაციური სქემების კანონზომიერებებს. მოსაზრება ამისშესახებ ჩამოაყალიბა 1937 წელს თავის სამაგისტრო ნაშრომში 21 წლის კლოდ შენონმა. შემდგომი 10 წლის განმავლობაში შენონი მუშაობდა საკითხებზე, რომლებიც 1948 წელს გამოაქვეყნა ინფორმაციის თეორიის სახით. შენონოს „აღმოჩენა“ ბულის ლოგიკისა და კომუტაციური ამოცანების თანხვდენაზე, შესაძლებელია ბულის დროცაც მომხდარიყო - კაკომუტაციო ელემენტები (ჩამრთველები და რელეები) იმ დროსაც ცნობილი იყო. შენონმა ეს მოახერხა, რადგან, მისივე თქმით, ერთდროულად ფლობდა მათემატიკასაც და ელექტროტექნიკასაც.

ბულის ალგებრა ოპერირებს მხოლოდ ორ ლოგიკურ მდგომარეობასთან (ლოგიკური ცვლადის მდგომარეობასთან): „ჭეშმარიტი“ ("TRUE") და „მცდარი“ ("TRUE"). ეს ორივე სიტყვიერი აღნიშვნა შეგვიძლია შევცვალით „1“-ით და „0“-ით, ან „დიახ“ და „არა“. ალბათ ადვილი მისახვედრია, რა ფართო ასპარეზი იქმნება ამ ლოგიკისათვის ორობით რიცხვით სისტემაში.

ბულის ალგებრის ლოგიკური ოპერაციები განსხვავდება ჩვეულებრივი არითმეტიკული ან ალგებრული ოპერაციებისაგან. მაგალითად, A + A = A და არა 2A. რა თქმა უნდა, ჩვენ ორი ჭეშმატიტი მდგომარეობა გვაქვს ფორმულის მარცხენა მხარეს, მაგრამ „+“ და „=“ სხვა შინაარსს ატარებს, და შედეგიც სხვა შინაარსს ატარებს.

ბულის ალგებრის განმარტებების და შედეგების განხილვა მეტად თვალსაჩინოა და გამარტივებულია ელექტრონული „ჩამკეტების“ მაგალითებზე.

**ლოგიკური ფუნქცია „და“ (AND Function)**

ლოგიკური ფუნქცია, ან ოპერატორი „და“ აფიქსირებს შედეგის სახით („1“ სახით გამოსავალზე), რომ ორი ან რამდენიმე მოვლენა მოხდა ერთად და ერთდროულად. ანუ ოპერატორის ყველა შესავალზე ერთად და ერთდროულად გაჩნდა ლოგიკური „1“. არა აქვს მნიშვნელობა როგორი თანმიმდევტობით დალაგებულია ეს მოვლენები ოპერატორის შესაცვლების მიმართ. ოპერატორისათვის A და B რაც B და A, ანუ A & B = B & A.

ელექტრონიკაშუი ფუნქციას წარმოადგენენ სიმბოლოებით ( . ) ან ( ). ამგვარად, A და B შესავლის მქონე „და“ ჩამკეტი გამოიხატება როგორც A**.**B, ან AB.

**ფუნქციის წარმოდგენა ჩამრთველებით**

|  |
| --- |
| logic and gate |

A და B ჩამრთველები იღებენ ლოგიკური „1“-ის მნიშვნელობას ჩართვის შემთხვევაში. ნათურა აინთება მხოლოდ იმ შემთხვევაში, თუ ორივე ჩამრთველი ჩართულია. გამოსავალი „ჭეშმარიტია“ (), თუ ორივე შესავალი „ჭეშმარიტია“. დააკვირდით, ელექტრულ ტერმინებში **„და“ ფუნქცია მიმდევრობით ჩართვას შეესაბამება**.

ჩამრთველების მდგომარეობის ოთხივე კომბინაცია და შედეგი აისახება ცხრილში:

**ჭეშმარიტების ცხრილი**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | A | B | გამოსავალი | აღწერა | | 0 | 0 | 0 | A და B გამორთულია, ნათურა **გამორთულია** | | 0 | 1 | 0 | A გამორთული, B ჩართული, ნათურა **გამორთულია** OFF | | 1 | 0 | 0 | A ჩართული, B გამორთული, ნათურა **გამორთულია** | | 1 | 1 | 1 | A ჩართული, B ჩართული, ნათურა **ჩართულია** | | ბულის გამოსახულება (A და B) | | | **A . B** | |
|  |
| 2-input AND Gate AND gate |

ლოგიკური ფუნქცია „ან“ **Tutorial: 2 of 7**

„ან“ ლოგიკური ფუნქციის შედეგი ჭეშმარიტია, თუ ჭეშმარიტია ერთი შესავალი მაინც.

ეს ფუნქცია ასრულებს ლოგიკური შეკრების ოპერაციას, რომელიც აღინიშნება სიმბოლოთი: A+B = Q.

**ფუნქციის წარმოდგენა ჩამრთველებით**

|  |
| --- |
| logic or gate |

ფუნქცია ხორციელდება ორი (ან მეტი) პარალელურად შეერთებული ჩამრთველით. ნათურა აინთება, თუ ერთი ჩამრთველი მაინც ჩართულია. **ელექტრონიკაში „ან“ ფუნქცია პარალელურ ჩართვას შეესაბამება.**

ჩამრთველების მდგომარეობის ოთხივე კომბინაცია და შედეგი აისახება ცხრილში:

**ჭეშმარიტების ცხრილი**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | A | B | გამოსავალი | აღწერა | | 0 | 0 | 0 | A და B გამორთულია, ნათურა **გამორთულია** | | 0 | 1 | 1 | A გამორთულია, B ჩართულია, ნათურა **ჩართულია** | | 1 | 0 | 1 | A ჩართულია, B გამორთულია, ნათურა **ჩართულია** | | 1 | 1 | 1 | A ჩართულია, B ჩართულია, ნათურა **ჩართულია** | | ბულის გამოსახულება (A ან B) | | | **A + B** | |
|  |
| 2-input OR Gate IEC OR gate symbol |

ლოგიკური ფუნქცია „არა“ **Tutorial: 3 of 7**

ეს უმარტივესი ლოგიოკური ფუნქცია ახდენს შესავალი ლოგიკური მდგომარეობის ინვერტყირებას, ამუ შეცვლას საწინააღმდეგო (ან დამატებით) მდგომარეობაზე. ეს ოპრეაცია აღინიშნება სიმბოლოთი ( ¯ ) - ხაზი სიმბოლოების ზემოდ.

**ფუნქციის წარმოდგენა ჩამრთველებით**

|  |  |
| --- | --- |
| logic not gate |  |

ფუნქცია წარმოდგენილია 2 ჩამრთველით. მათი მდგომარეომა საწინააღმდეგოა. თუ ჩამრთველი A გამორთულია, ნათურა ჩართულია . ამ დროს ჩამრთველი შესავლის საწინააღმდეგო მდგომარეობაშია და სქემაში უზრუნველყოფს ნათურის ჩართვას. თუ A ჩართულია, მისი უარყოფა გამორთავს ნათურას.

**ჭეშმარიტების ცხრილი**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | შესავალი | გამოსავალი | | 1 | 0 | | 0 | 1 | | ბულის გამოსახულება |  | |
|  |
| inverter IEC NOT gate symbol |

ინვერსია აღინიშნება ( O ) სიმბოლოთი შესავალზე ან გამოსავალზე. ცხადია

;

თუ

ინვერტორები შეგვიძლია შევქმნათ უკვე ცნობილი ელემენტებისაგან:

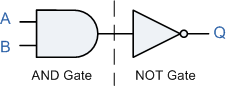
|  |
| --- |
| NAND or NOR Gate Inverter |

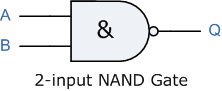
|  |
| --- |
|  |

ლოგიკური ფუნქციები „არა-და“ , „არა-ან“

**ფუნქცია „არა-და“**

ეს ფუნქცია შეგვიძლია შევქმნათ „და“ ფუნქციის უატყოფით:





„და“ ფუნქციის საწინააღმდეგოდ, გამოსავალზე გვაქვს „1“, როდესაც ორივე შესავალზე არის „0“.

**ფუნქციის წარმოდგენა ჩამრთველებით**

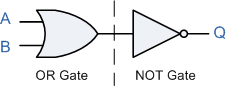
|  |
| --- |
| logic NAND gate |

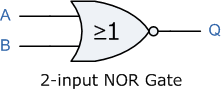
**ჭეშმარიტების ცხრილი**

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | გამოსავალი | აღწერა |
| 0 | 0 | 1 | A and B are both open, lamp ON |
| 0 | 1 | 1 | A is open and B is closed, lamp ON |
| 1 | 0 | 1 | A is closed and B is open, lamp ON |
| 1 | 1 | 0 | A is closed and B is closed, lamp OFF |
| ბულის გამოსახულება (A AND B) | | | **A . B** |

**„არა-ან“ ფუნქცია**

ეს ფუნქცია „ან“ ფუნქციის უარყოფას წარმოადგენს :





**ფუნქციის წარმოდგენა ჩამრთველებით**

|  |
| --- |
| logic NOR gate |

ერთი ჩამრთველის მაინც ჩართულობის შემთხვევაში დენი გადია ამ შემოკლებულ წრწეში და ნათურა ქრება.

**ჭეშმარიტების ცხრილი**

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | გამოსავალი | აღწერა |
| 0 | 0 | 1 | Both A and B are open, lamp ON |
| 0 | 1 | 0 | A is open and B is closed, lamp OFF |
| 1 | 0 | 0 | A is closed and B is open, lamp OFF |
| 1 | 1 | 0 | A is closed and B is closed, lamp OFF |
| ბულის გამოსახულება (A OR B) | | | **A + B** |

ბულის ალგებრის კანონები **Tutorial: 5 of 7**

ლოგიკური სიმბოლო „0“ და „1“ შეგვიძლია გამოვიყენოთ გამორთული ან ჩართლი კონტაქტის აღსანიშნავად. ბულის ალგებრის წესები და კანონები გვაძლევენ საშუალებას შევქმნათ და დავხვეწოთ რთული ლოგიკური სქემები.

ბულის ალგებრა წარმოადგენს ლოგიკაზე აგებულ მათემატიკურ სისტემას, რომლის საშუალებით შესაძლებელია შეიქმნას, გარდაიქმნას ან გამარტივდეს ბულის გამოსახულება. ბულის გამოსახულებაში შეგვიძლია მრავალი ცვლადი გამოვიყენოთ. ყოველ ცვლადს თავის ფუნქციონალური დატვირთვა ექნება, მაგრამ ცვლადი მხოლოდ ორ ლოგიკურ მნიშვნელობას იღებს. მაგალითად A, B, C ცვლადები ქმნიან ბულის გამოსახულებას A + B = C, მაგრამ ცვლადები მხოლოდ ორ ლოგიკურ მნიშვნელობას იღებენ.

ცხრილში მოცემულია ბულის ალგებრის წესები და თეორემები.

**ბულის ალგებრის ჭეშმარიტების ცხრილები**

|  |  |  |  |
| --- | --- | --- | --- |
| ფორმულა | განმარტება | სქემა | წესი ან კანონი |
| A + 1 = 1 | A in parallel with closed = "CLOSED"  A პარალელურად „ჩართულთან“ = “ჩართულს“ | universal parallel circuit | Annulment  გაუქმება |
| A + 0 = A | A in parallel with open = "A"  A პარალელურად „გამორთულთან“ = "A" | universal parallel circuit | Identity  იგივობა |
| A . 1 = A | A in series with closed = "A"  A მიმდევრობით ჩართულთან = "A" | universal series circuit | Identity  იგივობა |
| A . 0 = 0 | A in series with open = "OPEN"  A მიმდევრობით გამორთულთან = "გამორთულს" | universal series circuit | Annulment  გაუქმება |
| A + A = A | A in parallel with A = "A"  A და პარალელური A = "A" | indempotent parallel circuit | Indempotent  დუბლირება |
| A . A = A | A in series with A = "A"  A და მიმდევრობით A = "A" | indempotent series circuit | Indempotent  დუბლირება |
| NOT = A | NOT NOT A (double negative) = "A"  A -ს ორმაგი უარყოფა = "A" |  | Double Negation  ორმაგი უარყოფა |
| A + = 1 | A in parallel with not A = "CLOSED"  A და პარალელური = "ჩართული" | complement parallel circuit | Complement  დამატება |
| A . = 0 | A in series with not A = "OPEN"  A და მიმდევრობით = "გამორთული" | complement series circuit | Complement  დამატება |
| A+B = B+A | A in parallel with B = B in parallel with A  A პარალელურად B =  B პარალელურად A | absorption parallel circuit | Commutative  კომუტაცია |
| A.B = B.A | A in series with B = B in series with A  A მიმდევრობით B = B მიმდევრობით A | absorption series circuit | Commutative  კომუტაცია |
| = . | invert and replace OR with AND  ინვერტირება ცვლის „ან“-ს „და“-ზე |  | de Morgan's Theorem  დე მორგაბნის თეორემა |
| . = | invert and replace AND with OR  ინვერტირება ცვლის „და“-ს „ან“-ზე |  | de Morgan's Theorem  დე მორგანის თეორემა |

ბულის ალგებრის კანონების დაცვით შესაძლებელია გამოსახულებებზე ალგებრული ოპერაციების წარმოება და ნამრავლებათ დაშლა. ცხრილში მოყვანილი ოპერაციები განმარტებულია ორი ცვლადისათვის. ყველა მოყვანილი ოპერაცია ვრცელდება მრავალ ცვლადზე.

**კანონები და თეორემები**

ქვემოდ მოყვანილი თანაფარდობების გამოყენებით შესაძლებელია ლოგიკური ელექტრონული სქემების შექმნა და სრულყოფა.

* გაუქმების კანონი
  1. A . 0 = 0, A „და“ 0 ყოველთვის უდრის 0.
  2. A + 1 = 1, A „ან“ 1 ყოველთვის უდრის 1.
* იგივობის კანონი
  1. A + 0 = A, A „ან“ 0 თვით ცვლადის ტოლია.
  2. A . 1 = A, A „და“ 1 თვით ცვლადის ტოლია.
* დუბლირების კანონი
  1. A + A = A, A „ან“ A თვით ცვლადის ტოლია.
  2. A . A = A, A „და“ A თვით ცვლადის ტოლია.
* დამატების კანონი
  1. A . = 0, A „და“ მისი დამატება უდრის 0.
  2. A + = 1, A „ან“ მისი დამატება უდრის 1.
* კომუტატივობის კანონი
  1. A . B = B . A, „და“ ოპერაციის შესტულების დროს მნიშვნელობა არა აქვს ცვლადების მიმდევრობას.
  2. A + B = B + A, „ან“ ოპერაციის შესტულების დროს მნიშვნელობა არა აქვს ცვლადების მიმდევრობას.
* ორმაგი უარყოფის კანონი
  1. = A, A ცვლადის ორჯერ უარყოფა თვით ცვლადის ტოლია.
* დე მორგანის თეორემები
* (**1**) = .
* (**2**) = +

აგრეთვე:

* A . B . C = (A . B) . C = A . (B . C) გამრავლების ასოციატიური კანონი.
* A + B + C = (A + B) + C = A + (B + C) შეკრების ასოციატიური კანონი.
* A . (B + C) = A . B + A . C
* A + A . B = A
* A + B . C = (A + B) . ( A + C)
* 0 და 0 =

**მაგალითი No1**

მოყვანილი კანონების გამოყენებით გავამარტივოდ გამოსახულება: (A + B)(A + C)

|  |  |  |
| --- | --- | --- |
| Q = | (A + B)(A + C) |  |
|  | AA + AC + AB + BC | - Distributive law |
|  | A + AC + AB + BC | - Identity AND law (A.A = A) |
|  | A(1 + C) + AB + BC | - Distributive law |
|  | A.1 + AB + BC | - Identity OR law (1 + C = 1) |
|  | A(1 + B) + BC | - Distributive law |
|  | A.1 + BC | - Identity OR law (1 + B = 1) |
| Q = | A + BC | - Identity AND law (A.1 = A) |

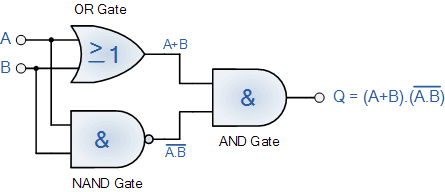
ამგვარად: (A + B).(A + C) = A + BC

**მაგალითი No2**

ლოგიკური სქემების სინთეზში ხშირად გამოიყენება „გამომრიცხავი ან“ ჩამკეტი მის მიერ შესრულებული ოპერაცია აღინიშნება სიმბოლოთი და წარმოდგენილია ჭეშმარიტების ცხრილით:

|  |  |  |  |
| --- | --- | --- | --- |
| სიმბოლო | ცხრილი | | |
| 2-input Ex-OR Logic Gate  **2-input Ex-OR Gate** | **B** | **A** | **Q** |
| **0** | **0** | **0** |
| **0** | **1** | **1** |
| **1** | **0** | **1** |
| **1** | **1** | **0** |
| ბულის გამოსახულება Q = AB | Read as A **OR** B but NOT **BOTH** gives Q | | |

ამ ჩამკეტის ექვივალენტური სქემა შესდგება სამი ჩამკეტისაგან. პირველი ასრულებს „ან“ ოპერაციას ცვლადებზე A და B. მეორე ასრულებს ოპერაციას „და-არა“, მესამე კი აერთიანებს ამ ჩამკეტების შედეგებს ოპერაციით „და“.



მაშასადამე ეს ლოგიკური ოპერაცია ბულის ალგებრაში წარმოდგენილია გამოსახულებით: AB = (A+B).()

ბულის ალგებრის ლოგიკური ოპერატორების ჭეშმარიტების ცხრილები **Tutorial: 6 of 7**

**საწყისი ოპერატორები:**

**„და“ ჩამკეტი 2 შესავლით - ლოგიკური გამრავლება**

|  |  |  |  |
| --- | --- | --- | --- |
| სიმბოლო | ცხრილი | | |
| 2-input AND Gate | A | B | Q |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |
| ბულის გამოსახულება Q = A.B | Read as A **AND** B gives Q | | |

**„ან“ ჩამკეტი 2 შესავლით - ლოგიკური შეკრება**

|  |  |  |  |
| --- | --- | --- | --- |
| სიმბოლო | ცხრილი | | |
| 2-input OR Gate | A | B | Q |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |
| ბულის გამოსახულება Q = A+B | Read as A OR B gives Q | | |

**ჩამკეტი „არა“ - ლოგიკური უარყოფა**

|  |  |  |  |
| --- | --- | --- | --- |
| სიმბოლო | ცხრილი | | |
| The NOT Gate | A | Q |  |
| 0 | 1 |  |
| 1 | 0 |  |
| ბულის გამოსახულება Q = NOT A or | Read as inverse of A gives Q | | |

**სინთეზირებული ოპერატორები:**

**ჩამკეტი „არა-და“ 2 შესავლით**

|  |  |  |  |
| --- | --- | --- | --- |
| სიმბოლო | ცხრილი | | |
| 2-input NAND Gate | A | B | Q |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |
| *ბულის გამოსახულება Q =* | *Read as NOT A or NOT B gives Q* | | |

***ჩამკეტი „არა-ან“ 2 შესავლით***

|  |  |  |  |
| --- | --- | --- | --- |
| სიმბოლო | ცხრილი | | |
| 2-input NOR Gate | A | B | Q |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |
| ბულის გამოსახულება Q = | Read as NOT A and NOT B gives Q | | |

**„გამომრიცხავი ან“ 2 შესავლით**

|  |  |  |  |
| --- | --- | --- | --- |
| სიმბოლო | ცხრილი | | |
| 2-input Ex-OR Gate IEC EX-OR gate symbol | A | B | Q |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |
| ბულის გამოსახულება Q = A⊕B |  | | |

**„გამომრიცხავი არა-ან“ 2 შესავლით**

|  |  |  |  |
| --- | --- | --- | --- |
| სიმბოლო | ცხრილი | | |
| 2-input Ex-NOR GateIEC EX-NOR gate symbol | A | B | Q |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |
| ბულის გამოსახულება Q = |  | | |

**2 შესავლის მქონე ჩამკეტების ცხრილი:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| შესავალი | | ჩამკეტების გამოსავალი | | | | | |
| A | B | AND | NAND | OR | NOR | EX-OR | EX-NOR |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |

.

ბულის ალგებრის გამოყენების მაგალითები **Tutorial: 7 of 7**

**მაგალითი No 1**

შეადგინეთ ჭეშმარიტების ცხრილი ლოგიკური ფუნქციებისათვის წერტილებში C, D და Q. რა ჩამკეტით შეიძლება შეიცვალოს წარმოდგენილი წრედი.

|  |
| --- |
| Example Circuit No1 |

ამ სქემის ჭეშმარიტების ცხრილი ასე გამოიყურება:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| შესავალი | | გამოსავალი | | |
| A | B | C | D | Q |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |

შევადგინოთ ამ სინთეზირებული ჩამკეტის ბულის გამოსახულება:

|  |  |  |
| --- | --- | --- |
| C =  D =  Q = | = | გამოყენებული წესი |
|  | = | AB = (A+B).() |
|  | = | AB = (A+B).() |
|  | = | = + |
|  | = |  |
|  | = |  |
|  | = | = + |
|  | = | A. = 0 |
|  | = |  |
|  | = | A + BC =(A + B)(A + C) |
| Q = |  |  |

|  |  |  |
| --- | --- | --- |
| Example Circuit No1 | ექვივალენტი | traditional OR gate symbol |

**მაგალითი No 2**

ოპოვეთ ბულის გამოსახულება ჩამკეტთა სისტემისათვის:

|  |
| --- |
| Example Circuit No2 |

შესავალი სიგნალებისათვის „და“ ჩამკეტი გვაძლევს შედეგს A.B, ხოლო „ან-არა“ ჩამკეტი . მათი გაერთიანება „ან“ ჩამკეტის საშუალებით გვაძლევს:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Example No2   |  |  |  |  |  | | --- | --- | --- | --- | --- | | შესავალი | | შუალედური ოპერაცია | | შედეგი | | B | A | A.B | A + B | Q | | 0 | 0 | 0 | 1 | 1 | | 0 | 1 | 0 | 0 | 0 | | 1 | 0 | 0 | 0 | 0 | | 1 | 1 | 1 | 0 | 1 | |

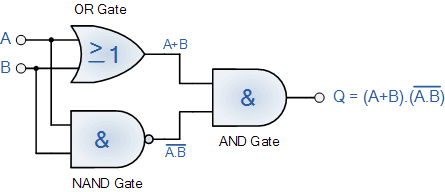
ამ გამოსახულების მეორე წევრი, დე მორგანის თეორემის თანახმად, საბოლოოდ გვაძლევს გამოსახულებას . ეს გამოსახულება შეესაბამება განხილულ ჩამკეტს „გამომრიცხავი არა-ან“ 2 შესავლით:

|  |  |  |  |
| --- | --- | --- | --- |
| სიმბოლო | ცხრილი | | |
| 2-input Ex-NOR GateIEC EX-NOR gate symbol | A | B | Q |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |
| ბულის გამოსახულება Q = |  | | |

ბულის გამოსახულებიდან ჩანს, რომ შესაძლებელია ამ ჩამკეტის სხვა ტექნიკური გადაწყვეტილებაც. შეგვიძლია „ან“ ჩამკეტით გავაერთიანოთ ორი „და“ ჩამკეტი, მხოლოდ ერთის ორივე შესავალი უნდა იყოს ინვერტირებული.

**მაგალითი No 3**

ჩამკეტი „გამომრიცხავი ან“ წარმოდგენილია ჩამკეტთა სისტემით:



შესაძლებელია მისი ბულის გამოსახულების გარდაქმნა და სისტემის სხვა ჩამლეტებით წარმოდგენა. დე მორგანის თეორემის გამოყენებით ვიღებთ: . შეგვიძლია „და“ ჩამკეტით გავაერთიანოთ ორი „ან“ ჩამკეტი, მხოლოდ, ამ შემთხვევაშიც, ერთის ორივე შესავალი უნდა იყოს ინვერტირებული.

**მაგალითი No 4**

იპოვეთ სისტემის ბულის გამოსახულება სქემისათვის:

|  |
| --- |
| Example Circuit No3 |

თითოეული ჩამკეტის გამოსავალზე თანმიმდევრულად მივიღებთ ბულის გამოსახულებებს:

|  |
| --- |
| Example No3 Answer |

ამ სისტემის ჭეშმარიტების ცხრილი გამოიყურება შემდეგნაირად:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| შესავალი | | | შუალედური შედეგი | | | | | გამოსავალი |
| C | B | A | A.B.C | B | C | B+C | A.(B+C) | Q |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |

მიღებული გამოსახულება შესაძლებელია გარდაიქმნას დე მორგანის თეორემის გამოყენებით: . ანუ ირკვევა, რომ მოქმედებს მხოლოდ ერთი შესავალი და სქემა ბუფერის ექვივალენტურია, ანუ უნდა შეიცვალის ბხოლოდ ერთი ჩამკეტით.

ეს მაგალითი გვიჩვენებს, რომ ბულის ალგებრა სისტემების გამარტივების და ოპტიმიზაციის საშუალებას წარმოადგენს.

**მაგალითი No 5**

7-სეგმენტიანი ციფრული ინდიკატორის დეშიფრატორი წარმოდგენილია ჭეშმარიტების ცხრილით:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | DEC | BIN | A | B | C | D | E | F | G |
| **7_segment_display_labeled**  **791px-Seven_segment_02_Pengo** | 0 | 0000 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0001 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 2 | 0010 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 3 | 0011 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| 4 | 0100 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 5 | 0101 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 6 | 0110 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 7 | 0111 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 8 | 1000 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 9 | 1001 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |

შევადგინოთ ჩამკეტების სისტემა, რომელიც უნდა მართავდეს ინდიკატორის სეგმენტებს. გვაქვს 4 შესავალი, რომელიც შეესაბამება ორობითი რიცხვების თანრიგებს. შესავლის კოდი უნდა მართავდეს სეგმენტებს. სათითაოთ განვიხილოთ სეგმენტები.

**სეგმენტი A**

თუ დავაკვირდებით ამ სეგმენტის სვეტს, დავინახავთ, რომ სეგმენტი ირთვება ყველა ათობითი რიცხვისათვის გარდა რიცხვებისა 1 და 4. სათანადო ორობითი კოდებია, ან შესავლების მდგომარეობებია 0001 და 0100. ავღნიშნოთ თანრიგები ცვლადებით .

პირველი რიცხვის შემთხვევაში გვაქვს ლოგიკური კომბინაცია: . ეს 4 შესავლის მქონე ლოგიკური ჩამკეტია, რომელმაც უნდა გამორთოს A სეგმენტი, როდესაც იღებს მნიშვნელობას 1, დანარჩენები კი მნიშვნელობას 0.

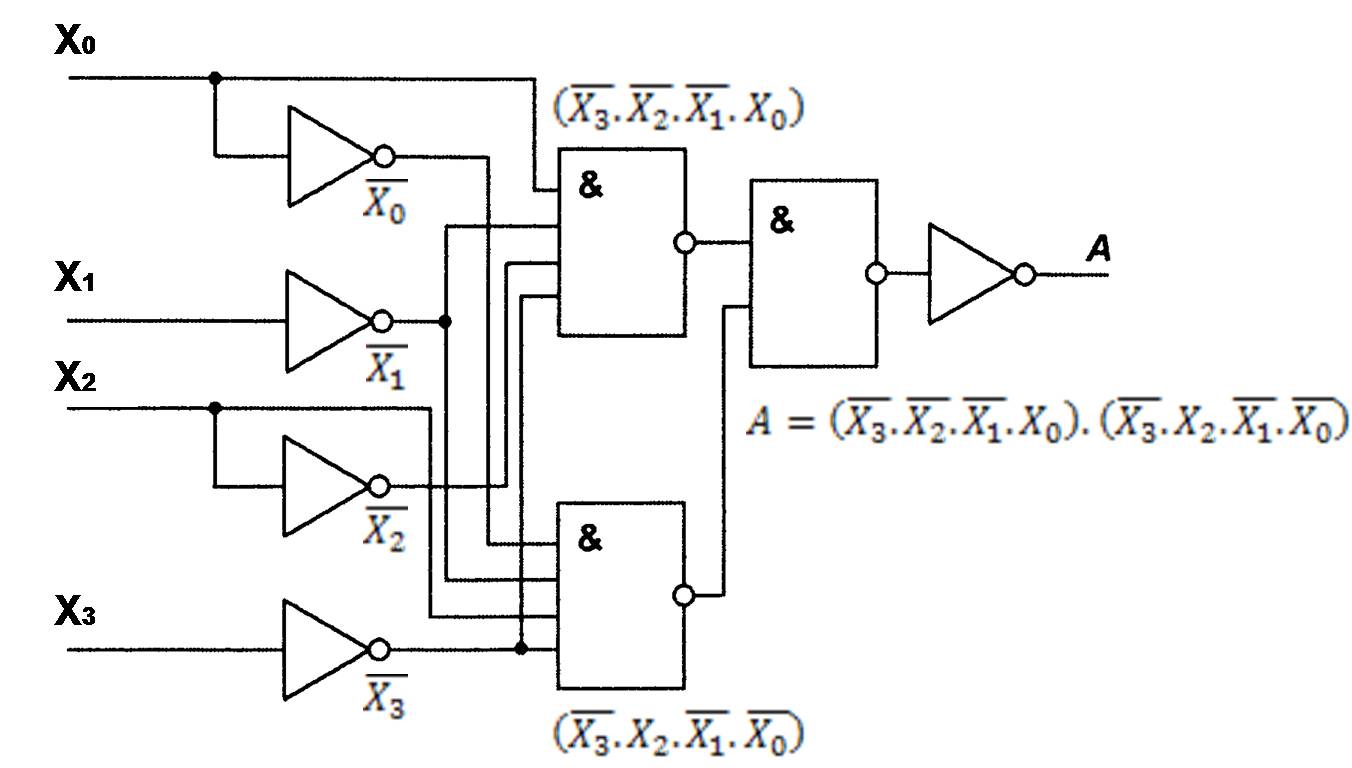
მეორე რიცხვის შემთხვევაში გვაქვს ლოგიკური კომბინაცია: ესეც 4 შესავლის მქონე ლოგიკური ჩამკეტია, რომელმაც უნდა გამორთოს A სეგმენტი, როდესაც იღებს მნიშვნელობას 1, დანარჩენები კი მნიშვნელობას 0.

სეგმენტის გამორთვა ნიშნავს ლოგიკურ ოპერაციას . სეგმენტი გამოირთვება **ან** პირველი **ან** მეორე ლოგიკური კომბინაციის შემთხვევაში, ე.ი. ორივე კომბინაცია უნდა გაერთიანდეს ლოგიკური ოპერატორით „ან“. ეს ნიშნავს, რომ 2 ნახსენები „და“ ჩამკეტი უნდა გაერთიანდეს „ან“ ჩამკეტით. ბულის გამოსავულება იქნება:

.

ამ გამოსახულების ექვივალენტია დე მორგანის ტეორემით გარდაქმნილი გამოსახულება . გამაერთიანებელი „ან“ ჩამკეტი შეცვლილია „და“ ჩამკეტით და ინვერტორით. საბოლოოდ ვიღებთ გამოსახულებას:

ამ გამოსახულებას შეესაბამება სქემა:



**საშინაო დავალება**

**მოცემულია ბულის ალგებრის წესების და 7-სეგმენტიანი ინდიკატორის მდგომარეობათა ცხრილი:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | 1 | A . 0 = 0; | A + 1 = 1 | | 2 | A + 0 = A; | A . 1 = A | | 3 | A + A = A; | A . A = A | | 4 | A . = 0; | A + = 1 | | 5 | A . B = B . A; | A + B = B + A | | 6 | = A, |  | | 7 | * = .; | = + | | 8 | A . B . C = (A . B) . C = A . (B . C) | | | 9 | A + B + C = (A + B) + C = A + (B + C) | | | 10 | A . (B + C) = A . B + A . C | | | 11 | A + A . B = A | | | 12 | A + B . C = (A + B) . ( A + C) | | | 13 | AB = (A+B).() | | | |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | DEC | BIN | A | B | C | D | E | F | G | | **7_segment_display_labeled** | 0 | 0000 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | | 1 | 0001 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | | 2 | 0010 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | | 3 | 0011 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | | 4 | 0100 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | | 5 | 0101 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | | 6 | 0110 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | | 7 | 0111 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | | 8 | 1000 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | 9 | 1001 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | |

**დავალება: შეადგინეთ ბულის გამოსახულება და ელექტრული სქემა ინდიკატორის სეგმენტებისათვის.**

**საგამოცდო ტესტის ამოცანები:**

ბულის გამოსახულების და სქემის შედგენა E სეგმენტისათვის. გამოიყენეთ 4 ან 2 შესავლის მქონე „და “ ჩამკეტები და ინვერტორები.

ბულის გამოსახულების და სქემის შედგენა F სეგმენტისათვის. გამოიყენეთ 4 ან 2 შესავლის მქონე „და “ ჩამკეტები და ინვერტორები.

ბულის გამოსახულების და სქემის შედგენა C და D სეგმენტისათვის. გამოიყენეთ 4 ან 2 შესავლის მქონე „და “ ჩამკეტები და ინვერტორები.

ბულის გამოსახულების და სქემის შედგენა C და G სეგმენტისათვის. გამოიყენეთ 4 ან 2 შესავლის მქონე „და “ ჩამკეტები და ინვერტორები.

ბულის გამოსახულების და სქემის შედგენა A და B სეგმენტისათვის. გამოიყენეთ 4 ან 2 შესავლის მქონე „და “ ჩამკეტები და ინვერტორები.

**დანართი**

**7-სეგმენტიანი ინდიკატორის დეშიფრატორის სრული სქემა**

