ALGJEBRA LLOGJIKE

HYRJE

Llogjika binare merret me variablat qe marrin vlera diskrete dhe me operatore qe marrin kuptimin llogjik. Ndersa çdo element llogjik(variabel) ose kusht duhet gjithmone te kete nje vlere llogjike (0 ose 1), gjithashtu duhet te ekzistojne menyra per te kombinuar sinjale ose kushte te ndryshme llogjike per te siguruar nje rezultat llogjik. Per shembull, merrni parasysh deklaraten llogjike: "Nese une e ndez celesin qe ndodhet tek muri, drita do te ndizet". Ne shikim te pare, kjo duket si nje deklarate e sakte. Megjithate, nese shohim disa faktore te tjere, kuptojme se ka me shume se aq. Ne kete shembull, nje deklarate me e plote do te ishte: "Nese une e ndez celesin dhe llamba eshte funksionale dhe elektriciteti eshte prezent, drita do te ndizet". Nese i shohim keto dy deklarata si shprehje llogjike dhe perdorim terminologjine llogjike, mund te formulojme deklaraten e pare ne:

$$Light = Switch$$

Kjo nuk do te thote asgje me shume se drita do te ndjeke veprimin Switch, keshtu qe kur celesi te jete lart (ON / true / 1) drita do te jete gjithashtu ON / true / 1. Ne anen tjeter, nese celesi eshte poshte (OFF / false / 0), drita do te jete gjithashtu OFF / false / 0. Duke pare versionin e dyte te deklarates, shprehja e formuluar do te ishte pak me komplekse:

$$Light = Switch\ AND\ Bulb\ AND\ Power$$

OPERATORET DHE PORTAT LLOGJIKE KRYESORE

Kur kemi te bejme me qarqet llogjike (si ne kompjutera), ne jo vetem qe duhet te merremi me funksionet llogjike; por gjithashtu kemi nevoje per disa simbole te veçanta per te percaktuar keto funksione ne nje skeme llogjike. Ekzistojne tri operacione themelore llogjike, nga te cilat mund te rrjedhin te gjitha funksionet e tjera, sado komplekse. Keto funksione jane emeruar **AND**, **OR**, dhe **NOT**. Secila prej tyre ka nje simbol te veçante dhe nje sjellje te percaktuar qarte.

Blloqet themelore qe formojne nje kompjuter (fizikisht) quhen **porta llogjike** ose vetem porta. Portat jane qarqe baze qe kane te pakten nje (dhe zakonisht me shume) hyrje dhe saktesisht nje dalje. Vlerat e hyrjes dhe te daljes jane vlerat llogjike TRUE (1) dhe FALSE (0). Ne arkitekturen kompjuterike eshte e zakonshme te perdorni 0 per false dhe 1 per true. Portat nuk kane kujtese. Vlera e daljes varet vetem nga vlera aktuale e hyrjeve. Nje menyre e dobishme per te pershkruar marredhenien ndermjet vlerave hyrese te portave dhe daljes eshte tabela e se vertetes. Ne nje tabele te se vertetes, vlera e çdo outputi eshte tabeluar per çdo kombinim te mundshem te vlerave te hyrjes. Tre portat llogjike kryesore si dhe tre veprimet kryesore llogjike jane AND, OR dhe NOT. AND, OR dhe NOT ndryshe quhen dhe **operatore llogjike**.

a. Porta AND

Operatori AND		
X	**Y**	**Z=X*Y**
0	0	0
0	1	0
1	0	0
1	1	1

Fig 1 (Tabela e vertetesise)

Skematikisht porta llogjike AND shprehet si me poshte:

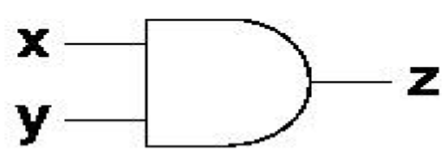


Fig 2 (Porta AND)

Nuk ka kufi per numrin e inputeve qe mund te aplikohen ne nje porte AND. Sidoqofte, per arsye praktike, portat AND me se shumti prodhohen me 2, 3 ose 4 inpute.

b. Porta OR

Operatori OR		
X	**Y**	**Z=X+Y**
0	0	0
0	1	1
1	0	1
1	1	1

Fig 3 (Tabela e vertetesise)

Skematikisht porta llogjike OR shprehet si me poshte:

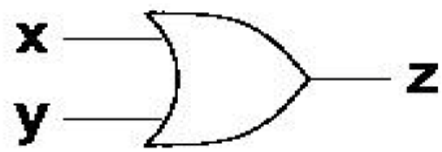


Fig 4 (Porta OR)

Ashtu si dhe porta AND, as porta OR nuk ka kufi per numrin e inputeve qe mund te aplikohen. Sidoqofte, per arsye praktike, portat OR me se shumti prodhohen me 2, 3 ose 4 inpute.

c. Porta NOT

Operatori NOT	
X	$\bar{X}$
0	1
1	0

Fig 5 (tabela e vertesise)

Skematikisht porta llogjike NOT shprehet si me poshte:

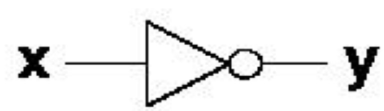


Fig 6 (Porta NOT)

PORTAT LLOGJIKE TE KOMBUNUARA

Ndonjehere lind nevoja e kombinimit te portave baze per te formuar porta me komplekse. Me poshte paraqiten portat e kombinuara skematikisht se bashku me tabelen e vertetesise:

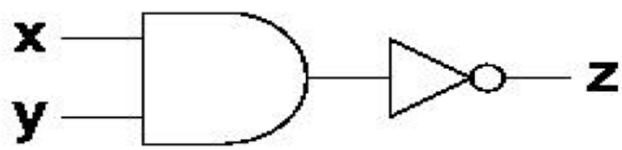
a. Porta NAND

Porta NAND eshte e formuar nga nje porte AND dhe nje porte NOT.

Operatori NAND		
X	**Y**	**Z=**$\overline{X * Y}$
0	0	1
0	1	1
1	0	1
1	1	0

Fig 7 (Tabela e vertetesise)

Ashtu si porta AND, edhe porta NAND mund te kete disa inpute. Tabela e vertetesise per porten NAND eshte njesoj si tabela e portes AND ku outputet jane te invertuara. NAND ndryshe quhet komplementi i AND. Me poshte paraqitet skema e portes NAND:



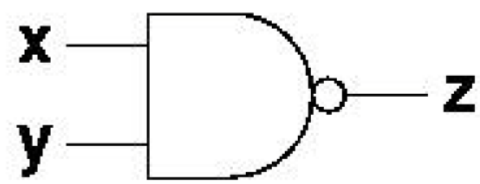


Fig 8 (Porta NAND)

a. Porta NOR

Porta NOR eshte e formuar nga nje porte OR dhe nje porte NOT.

Operatori NOR		
X	**Y**	**$Z=\overline{X+Y}$**
0	0	1
0	1	0
1	0	0
1	1	0

Fig 9 (Tabela e vertetesise)

Ashtu si porta OR, edhe porta NOR mund te kete disa inpute. Tabela e vertetesise per porten NOR eshte njesoj si tabela e portes OR ku outputet jane te invertuara. NOR ndryshe quhet komplementi i OR. Me poshte paraqitet skema e portes NOR:

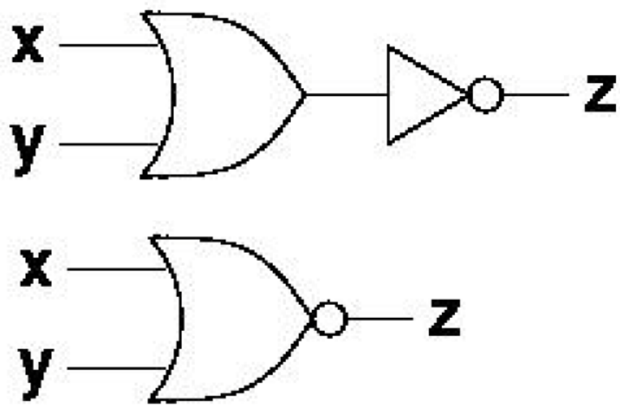


Fig 10 (Porta NOR)

a. Porta XOR

Porta XOR eshte e ngjashme me porten OR. Ashtu si porta OR, edhe porta XOR mund te kete disa inpute. Outputi i portes XOR eshte 1 nese ka nje dhe vetem nje 1 si input nga te gjitha inputet qe ka porta(inputet e tjere jane te gjithe 0).

Operatori XOR		
X	**Y**	**Z=X $\oplus$ Y**
0	0	0
0	1	1
1	0	1
1	1	0

Fig 11 (Tabela e vertetesise)

Me poshte paraqitet skema e portes XOR:

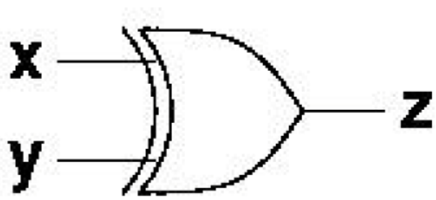


Fig 12 (Porta XOR)

Portat jane themelet e ndertimit te nje qarku llogjik. Prodhimi fizik i tyre eshte i shumellojshem, keshtu mund te permendim: portat llogjike te ndertuara me dioda(DL), portat llogjike te ndertuara me tranzistore dhe rezistenca(RTL), portat llogjike te ndertuara me dioda dhe tranzistore(DTL), portat CMOS, etj. Te gjitha keto lloje portash kane avantazhet dhe disavantazhet e tyre qofte nga ana ekonimike ashtu edhe nga ana funksionale. Ne saje te problematikes qe has nje inxhinier, ben perzgjedhjen e portes se duhur per te prodhuar qarqet llogjike.

ALGJEBRA LLOGJIKE (BOOL-eane)

Nje nga qellimet kryesore kur ndertohen qarqet digjitale eshte gjetja e menyrave per t'i bere ato sa me te thjeshta qe te jete e mundur. Kjo vazhdimisht kerkon qe shprehjet komplekse logjike te reduktohen ne shprehje me te thjeshta te cilat prodhojne rezultate te njejta ne te gjitha kushtet e mundshme. Shprehja me e thjeshte pastaj mund te zbatohet me nje qark me te vogel dhe me te thjeshte, i cili nga ana tjeter kursen çmimin e portave te panevojshme, zvogelon numrin e portave te nevojshme dhe zvogelon fuqine dhe sasine e hapesires se kerkuar nga ato porta. Qarqet e thjeshta jane me te lira per t'u prodhuar, shpenzojne me pak energji dhe punojne me shpejte se qarqet komplekse.

Nje mjet per te reduktuar shprehjet logjike eshte matematika e shprehjeve logjike, te paraqitura nga George Boole ne 1854 dhe i njohur sot si algjebra llogjike (Boolean Algebra). Rregullat e algjebres Boolean-e jane te thjeshta dhe te drejtperdrejta, dhe mund te aplikohen per çdo shprehje logjike. Shprehja e thjeshtuar qe del nga aplikimi i rregullave te algjebres llogjike, mund te testohet kollaj nese jep te njejtin output me shprehjen e pathjeshtuar, duke perdorur nje tabele vertetesie per cdo kombinim inputesh.

Algjebra e Bulit eshte sistem matematikor per manipulim me ndryshore qe mund te kene njeren nga dy vlera te mundshme (0 ose 1). Shprehjet Buleane (logjike) fitohen duke kryer veprime me operatoret llogjike. – Veprimet e zakonshme jane: NOT,AND dhe OR .

Ne nje shprehje Bool-eane perparesite e veprimeve jane si me poshte:

1. Kllapat
2. Veprimi NOT
3. Veprimet AND dhe OR

Algjebra llogjike (Bool-eane) eshte nje strukture algjebrike e percaktuar ne nje bashkesi B elementesh ku kane kuptim dy veprime (+, *) dhe rregullat e meposhtme:

1. Veprimet + dhe * jane injektive ne bashkesine B.
2. Per cdo element X nga bashkesia B, ekziston elementi 0 i tille qe X+0=0+X=X Dhe elementi 1 i tille qe X*1=1*X=X
3. Per cdo X dhe Y nga bashkesia B, X+Y=Y+X dhe X*Y=Y*X
4. Per cdo X, Y dhe Z nga bashkesia B, X*(Y+Z)=X*Y+X*Z dhe X+(Y*Z)=(X+Y)*(X+Z)
5. Per cdo element X nga B, ekziston komplementi i tij $\overline{\boldsymbol{X}}$ i tille qe X+$\overline{\boldsymbol{X}}$ = 1 dhe X * $\overline{\boldsymbol{X}}$ = 0

Ekzistojne shume identitete logjike(te cilat njihen si **Postulatet e Bulit** ose Rregullat e Bulit) te cilat na ndihmojne te bejme thjeshtimin e shprehjeve llogjike. Me poshte paraqiten rregullat:

POSTULATI i BULIT	FORMA AND	FORMA OR
Vetia e identitetit	X * 1 = X	X + 0 = X
Vetia e dominances	X * 0 = 0	X + 1 = 1
Vetia e vetvetes	X * X = X	X + X = X
Vetia e inversit	X * $\overline{X}$ = 0	X + $\overline{X}$ = 1
Vetia e nderrimit	X * Y = Y * X	X + Y = Y + X
Vetia e shoqerimit	A * (B * C) = (A * B) * C	A + (B + C) = (A + B) + C
Vetia e shperndarjes	X + Y*Z = (X + Y)*(X + Z)	X*(Y + Z) = X*Y + X*Z
Vetia e perthithjes	X*(X + Y) = X	X + X*Y = X
Vetia e perthithjes 2	X * ($\overline{X}$ + Y) = X * Y	X + ($\overline{X}$ * Y) = X + Y
Vetia e komplementit te dyfishte	$\overline{\overline{X}}$ = X	
Ligjet De Morgan	$\overline{A * B} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A} * \overline{B}$

Funksioni logjik (i Bulit) ka te pakten:

- Nje ndryshore logjike
- Nje veprim logjik
- Nje te dhene hyrese (input) nga bashkesia {0,1}
- Ai prodhon nje rezultat (output) po ashtu nga bashkesia {0,1}

Funksion Logjik F i n ndryshoreve binare $x_1, x_2 \ldots x_n$ quhet ligji sipas te cilit cdo njeri prej kombinimeve binare te n ndryshoreve i vihet ne korrenspodence nje vlere 0 ose 1. Funksioni mund te jepet ne forme tabelare algjebrike(tabele vertetesie), ose ne forme te rrjetit kombinator (ne porta llogjike) . Qarku llogjik kombinatorik perbehet nga porta logjike, rezultatet e te cilave(outputi) ne çdo kohe percaktohen direkt nga kombinimi i tanishem i inputeve pa marre parasysh hyrjen e meparshme. Nje qark llogjik kombinatorik kryen nje procesim specifik te informacionit, procesim i cili mund te shprehet llogjikisht nga nje funksion llogjik dhe operatore llogjike. Nje qark kombinator eshte nje porte e pergjithesuar. Ne pergjithesi nje qark i tille ka m hyrje dhe n dalje. Nje qark i tille gjithmone mund te ndertohet si bashkesi qarqesh kombinatore me te vogla, secila me saktesisht nje dalje. Per kete arsye, do te studiojme qarqet kombinatore me saktesisht nje dalje. Per te percaktuar menyren e sakte te funksionimit te nje qarku kombinator, mund te perdorim metoda te ndryshme, siç jane shprehjet logjike ose tabelat e vertetesise. Ashtu sic e permendem me siper, nje tabele vertetesise eshte nje prezantim i plote i te gjitha kombinimeve te mundshme te vlerave te hyrjes, secila e lidhur me vleren e daljes.

Me poshte paraqitet nje shembull zgjidhje te nje problemi me tabele vertetesie:

PROBLEMI

Tre celesa komandojne ndezjen e nje llambe. Duhet te pakten dy celesa te jene ON qe llamba te ndizet. Ne cdo rast tjeter llamba rri fikur.

ANALIZA

Duke qene se do te ndertohet nje qark llogjik kombinator, qarku do te shprehet me nje funksion llogjik duke perdorur algjebren e bulit. Atehere celesat do te perfaqesohen si vlera binare, gjithashtu dhe rezultati (celesi ON -> 1, celesi OFF-> 0 llamba ndezur -> 1, llamba fikur -> 0). Duke shenuar tre celesat me variablat x,y dhe z, lehtesisht mund te ndertohet tabela e vertetesise si me poshte:

x y z	**F**
0 0 0	0
0 0 1	0
0 1 0	0
0 1 1	1
1 0 0	0
1 0 1	1
1 1 0	1
1 1 1	1

Nje shprehje logjike mund te prezantohet ne forma te ndryshme:

- Keto forma jane logjikisht ekuivalente.
- Shprehjet logjikisht ekuivalente kane tabela vertetesie te njejta.
- Qe te eliminohet konfuzioni, funksionet logjike shprehen ne forme *kanonike* ose *standarde*.

Ekzistojne dy forma kanonike per shprehjet logjike:

- Shume prodhimesh - $\sum$ - merren vlerat e ndryshoreve ku rezultati eshte 1 (pra funksioni eshte i sakte). Quhen ndryshe **MINTERMA**
- Prodhim shumash - $\prod$ - merren vlerat e ndryshoreve kur rezultati eshte 0 (pra funksioni eshte i gabuar). Quhen ndryshe **MAKSTERMA**

x y z	F
0 0 0	0
0 0 1	0
0 1 0	0
0 1 1	1
1 0 0	0
1 0 1	1
1 1 0	1
1 1 1	1

maksterm

minterm

Nisur nga tabela e vertetesise mund te evidentojme funksionet kanonike:

Shume prodhimesh: $F = \bar{x} * y * z + x * \bar{y} * z + x * y * \bar{z} + x * y * z$

Prodhim shumash: $F = (x + y + z) * (x + y + \bar{z}) * (\bar{x} + y + z) * (x + \bar{y} + z)$

Sic u permend edhe me siper, postulatet e Bulit sherbejne per te thjeshtuar funksionin e nje qarku llogjik. Nese marrim shumen e prodhimeve:

$$F = \bar{x} * y * z + x * \bar{y} * z + x * y * \bar{z} + x * y * z$$

Per kete funksion mund te bejme thjeshtimet e meposhtme:

$F = \bar{X} * Y * Z + X * \bar{Y} * Z + X * Y * \bar{Z} + X * Y * Z$

$F = \bar{X} * Y * Z + X * \bar{Y} * Z + X * Y * \bar{Z} + X * Y * Z + X * Y * Z + X * Y * Z$ Vetia e vetvetes, shtojme dy here termin X*Y*Z

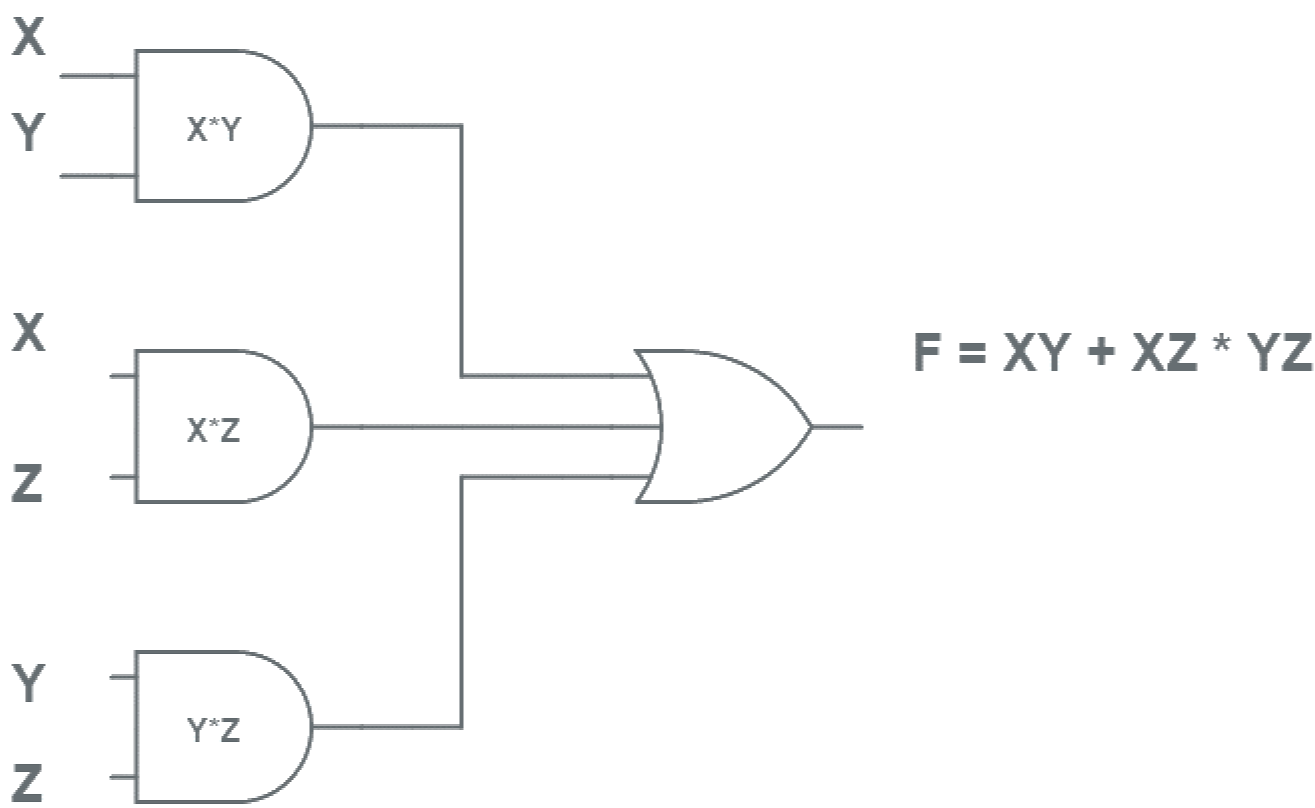
$F = Y * Z * (\bar{X} + X) + X * Z * (\bar{Y} + Y) + X * Y * (\bar{Z} + Z)$ Vetia e Shperndarjes

$F = Y * Z + X * Z + X * Y$ Vetia e Inversit dhe Vetia e Indentitetit

$F = X * Y + X * Z + Y * Z)$ Vetia e Nderrimit

Pra funksioni mund te thjeshtohet ne nje forme me pak te komplikuara. Funksionet logjike pasi jane thjeshtuar implementohen ne qarqe te kompjutereve digjital te quajtura porta (gates). Portat jane blloqet themelore te ndertimit per dizajn dixhital te pajisjeve elektronike. Sic e permendem me siper, porta eshte nje pajisje elektronike qe prodhon rezultat bazuar ne nje ose me shume vlera hyrese. Qarqet e integruara permbajne grupe te portave te ndertuara per qellime te ndryshme.

Per funksionin e thjeshtuar te problemit te mesiperm, skema e qarkut llogjik kombinatorik do te ishte si me poshte:



Kur ndertohen qarqet logjike duhet te kihet parasysh te perdoret nje numer minimal i portave logjike. Perdorimi i minimal i portave logjike paraqet perparesite e meposhtme:

- Rezultati merret me shpejt
- Ulet mundesia e gabimit gjate montimit
- Lehteson mirembajtjen
- Ulet kostoja e prodhimit

Me poshte, po analizojme nje shembull tjeter per te treguar thjeshtimin qe mund ti bejme nje funksioni llogjik duke perdorur postulatet e Bulit:

A eshte i vertete barazimi i meposhtem ?

$(A + \bar{A}B)(A + AB)(\bar{A} + B + \bar{C})$ = AB + A$\bar{C}$

Marrim shprehjen ne anen e majte dhe shumezojme dy kllapat e para:
$(A + \bar{A}B)(A + AB)$ = AA + AAB + $\bar{A}B$A + $\bar{A}B$AB

Nga postulatet e Bulit, A*A = A dhe $\bar{A} * A = 0$, atehere pas thjeshtimeve mbetet:

(AA + AAB + $\bar{A}B$A + $\bar{A}B$AB) = A + AB

(A + AB))$(\bar{A} + B + \bar{C})$ = A$\bar{A}$ + AB + A$\bar{C}$ + AB$\bar{A}$ + ABB + AB$\bar{C}$ = AB + A$\bar{C}$ + AB + AB$\bar{C}$

Mund te bejme nje faktorizim me AB:

AB*(1 + 1 + $\bar{C}$) + A$\bar{C}$

Shprehja 1 + 1+ $\bar{C}$ eshte e barabarte me 1. Me pas AB * 1 = AB dhe si perfundim shprehja ne te majte eshte AB + A$\bar{C}$ aq sa shprehja ne te djathte, pra barazimi eshte i vertete.