

(19)



(10) **LT IP384 A**

(12) **PARAIŠKOS APRAŠYMAS**

(21) Paraiškos numeris: **IP384**

(51) Int. Cl. (2006): **G06F 7/00**

(22) Paraiškos padavimo data: **1993 03 05**

G06F 7/38

(41) Paraiškos paskelbimo data: **1994 11 25**

G06F 7/48

G06F 9/302

G06F 9/44

(62) Paraiškos, iš kurios dokumentas išskirtas, numeris: —

G06F 12/00

G11C 11/412

(86) Tarptautinės paraiškos numeris: —

G11C 11/417

G11C 11/419

(86) Tarptautinės paraiškos padavimo data: —

G11C 15/00

(85) Nacionalinio PCT lygio procedūros pradžios data: —

(30) Prioritetas: **9002558, 1990 08 02, SE**
PCT/SE91/00515, 1991 08 01, SE

(71) Pareiškėjas:

CARLSTEDT ELEKTRONIK AB, Industrivagen 55, S-433 61 Partille, SE

(72) Išradėjas:

Lars Gunnar CARLSTEDT, SE

(74) Patentinis patikėtinis/atstovas:

Reda ŽABOLIENĖ, Advokatės Redos Žabalienės kontora METIDA, Verslo centras VERTAS, Gynėjų g. 16, LT-01109 Vilnius, LT

(54) Pavadinimas:

Būdas ir skaitmeninis aritmetinis įtaisas aritmetinėms, loginėms ir panašioms operacijoms atlikti

(57) Referatas:

—

BŪDAS IR SKAITMENINIS ARITMETINIS ĮTAISAS ARITMETINĖMS, LOGINĖMS IR PANAŠIOMS OPERACIJOMS ATLIKTI

Išradimas susijęs su aritmetinių, loginių ir panašių operacijų su skaitmeninio dydžio elementais atlikimo būdu ir aritmetiniu procesoriaus įtaisu.

Išradimo pagrindimas

Aritmetinis loginis įtaisas (ALĮ) - tai kompiuterio blokas, kuriame atliekamos su informacijos elementais aritmetinės, loginės ir panašios operacijos. Operacijos atliekamos panaudojant aritmetines komandas, paduodamas į ALĮ. Daugelis žinomų ALĮ yra sujungiami su akumuliatoriumi ir/arba buferiniais registrais arba kito tipo registrais, laikinai išsaugančiais duomenis.

IŠRADIMO TIKSLAI

Pagrindinis išradimo tikslas yra sukurti metodą arba aritmetinį loginį įtaisą (ALĮ), supaprastinantį skaitmenines išraiškas. Kitas išradimo tikslas yra sukurti ALĮ, kuris veikimo metu, nesinaudotų laikina atmintimi.

Dar kitas išradimo tikslas yra sukurti ALĮ, kuris naudoja aritmetines komandas pateiktai informacijai apdoroti.

Tolimesnis išradimo tikslas yra sukurti ALĮ, leidžiantį slankaus kablelio dydį atspindėti kaip operandą su keičiamu mantisės ir eksponentės padalinimu.

Tolimesnis išradimo tikslas yra sukurti ALĮ, leidžiantį valdyti bitų informaciją magistralėse su slankaus kablelio dydžiais jose kaip operandus ir kodo lauką, informuojantį apie padalinimą tarp mantisės ir eksponentės.

Dar tolimesnis išradimo tikslas sukurti ALĮ, leidžiantį valdyti, magistralėse esantiems operandams, duotas aritmetines komandas, paduodamas į keičiamo kodo ilgio kodo lauką.

Dar tolimesnis išradimo tikslas sukurti ALĮ, įvertinantį kovariaciją tarp slankaus kablelio ir sveiko skaičiaus interpretacijų. Kodo ilgis užduodamas.

Išradime pateiktas skaitmeninis aritmetinis įtaisas buvo kuriamas projektuojant naujos rūšies sumažinto instrukcijų skaičiaus procesorių. Struktūrinės aritmetikos operacijos tada buvo realizuotos aritmetiniame įrenginyje, skirtame struktūrinei aritmetikai, bet aritmetinės ir panašios operacijos buvo atliekamos skaitmeniniame aritmetiniame loginiame įtaise, skaitmeniniame ALĮ. Įprastinio tipo ALĮ gali būti valdomi taip, kad atliktų šias aritmetines ir panašias operacijas.

Patente US 4 615 003, priklausančiame Assignee Burroughs Corporation, aprašomas sumažinto instrukcijų skaičiaus procesorius, turintis registrų failą, kuriame lygiagretus informacijos perdavimas tarp registrų gaunamas optimizuojant dviejų mazgų grafą. Aritmetinis loginis įtaisas, poroje su registrų failu, vykdo paprastas aritmetines operacijas. Šis ALĮ nebus detalizuojamas tolimesniame aprašyme. Patente US 4 502 118 taip pat priklausančiame Assignee Burroughs Corporation, aprašomas sumažinto instrukcijų skaičiaus procesorius, turintis ALĮ, kuris atlieka standartines aritmetines operacijas ir Bulio logines operacijas. Šis ALĮ nebus detalizuojamas tolimesniame aprašyme.

Šie du dokumentai rodo, kad yra bendros aritmetinių loginių įrenginių panaudojimo sumažinto instrukcijų skaičiaus procesoriuose žinios.

IŠRADIMO REZIUOME

Tam, kad realizuoti nors pagrindinį išradimo tikslą, paminėtą aukščiau, išradime pateiktas bendras aritmetinių, loginių ir panašių operacijų su skaitmeninio vertės elementais atlikimo būdas apimantis sekančius žingsnius:

a) pateikimas įėjimo sąrašo, sudaryto iš skaitmeninio dydžio elementų, kurie bus apdorojami, ir komandos informacijos.

b) tiesioginis operacijos, su skaitmeninio dydžio elementais įėjimo sąraše, atlikimas ir valdymas, naudojant instrukcijų informaciją.

c) kiekvienas skaitmeninio dydžio elementas pateikiamas į žodį, turintį iš anksto apibrėžtą ilgį (bitų skaičių), didžiausias žodžių skaičius atitinka didžiausią pateikiamų elementų skaičių.

d) skaičiavimai atliekami, apdorojimo įtaisu, perrašant elementus į išėjimo sąrašą.

Optimaliausias (maksimalus) elementų įėjimo ir išėjimo sąraše skaičius - keturi. Vienas sąrašo elementas tada gali būti susijęs su komandos žodžiu, kiti du sąrašo elementai susiję su dviem atitinkamais skaitmeniniais dydžiais, ir vienas gali būti susietas su tarpiniu rezultatu, gautu skaičiavimo metu pakartojant įėjimo elementų perrašymą keletą kartų.

Sąrašas optimaliausiai susideda iš panaudojimo funkcijos, kurioje vienas iš elementų yra komandos kodas, o kiti likusieji yra komandos argumentai. Komanda vykdoma kartojant komandos kodo perrašymą iš išėjimo sąrašo į įėjimo sąrašą iki gaunamas galutinis rezultatas. Kiekvieno perrašymo pakartojimo metu, sąrašas gali būti perrašytas, įtraukiant po dydžio žodžio, jeigu to reikia atliekamiems skaičiavimams, pakeistą komandos kodo žodį.

Geriausia, prieš atliekamą apdorojimą, skaitmeninio dydžio žodžius paduoti koduotoje formoje. Tinkamiausia, kai skaitmeniniai dydžiai pateikiami perkoduoti į glaustesnį aprašymą, t.y. kai kiekvieną koduotą dydį atitinka tiksliai vienas interpretuojamas dydis. Pirmasis kodavimas gali būti atliktas sveiką dydį atspindint binariniu žodžiu. Antrasis kodavimas gali būti atliktas slankaus kablelio dydį atspindint binariniu žodžiu. Optimaliausias toks kodavimas, kad baigus kodavimą, slankaus kablelio dydžiai atspindimi tokiu pat būdu kaip ir sveiki dydžiai. Tokiu būdu, dydžiai gali būti pateikti ir ne dvejetainėje formoje. Aparatūra leidžia atlikti skaičiavimus su transformuotais bitų paternais tik besąlygiškai vykdant

užduotas taisyklės, t.y. nežinant kaip buvo interpretuojami dydžiai.

Paprasčiausias būdas atspindėti slankaus kabelio dydį, taip, kad gauti kintamą eksponentės plotį, yra į binarinį koduotą slankaus kabelio dydžio atspindėjimą įtraukti kodo, eksponentės ir mantinės laukus, kai kodo lauke nurodoma eksponentės ir mantinės kintančio ilgio laukų riba. Sutankintas atspindėjimas gali būti gautas įjungiant virtualų „1“, t.y. „1“, kuris fiziškai neatspindimas, slankaus kabelio dydžio mantinės lauko žodžio priekyje.

Žemiau seka išraiškų, naudojamų šiame aprašyme sąrašas ir jų papildomos reikšmės:

uždarumas hierarchiškai struktūrizuota būtis, kuri apibrėžia procesą. Visi uždaramai turi šaknį, kuri vieninteliu būdu apibrėžia uždaramą. Supaprastinimas sumažinto instrukcijų skaičiaus mašinose atliekamas naudojant uždaramus. Visos mašinos būsenos gaunamos supaprastinimų dėka.

tikslas	turimas vykdyti
uždarumas,	t.y. supaprastinamas.
grupė	grupė sudaryta iš keturių dvejetainių skaičių. Kiekvienas žodis susideda iš eilės grupių.
išpjova	vieno arba kelių žodžių dalis. Dalis sudaryta iš dviejų bitų gali, pavyzdžiui, susidėti iš dviejų žodžių mažiausiai reikšmingų bitų.
išlyginti	įsitikinti, kad naudojamas bitas yra teisingoje žodžio vietoje.
perteklius	duomenys arba bitas, kurie pridedami tam, kad nustatyti klaidą.
normalizuotas kodas	slankaus kabelio dydis, kuris yra atspindimas tokiu būdu, kad reikšmingiausias mantinės bitas yra virtualus „1“ - teigiamiesiems dydžiams ir „0“ - neigiamiesiems.
binarinis atspindėjimas	„0“ ir/arba „1“ kombinacija, atspindinti bet kurios skaičiavimo sistemos skaičių.
dvejetainis skaičius	sistemos, kurios pagrindas 2 skaičius.
H - kodas	specialus kodas, naudojamas išradime pateiktame ALI, skaičių atspindėjimui. Taip pat vadinamas H - kodu.
sveikas H-dydis	specialus kodas sveikiems dydžiams atspindėti.
slankaus kabelio H-dydis	specialus kodas slankaus kabelio dydžiams atspindėti.
sutankintas atspindėjimas	kiekviena binarinė kombinacija (bitų paternas) atitinkantis vienintelį dydį.

TRUMPAS PAVEIKSLŲ APRAŠYMAS

Pilnesniam išradimo aprašymo, tolimesnių tikslų ir pranašumų supratimui, pateikiami sekantys surišti su pridėtais paveikslais aprašymai, kuriuose:

- Fig. 1A - išradime pateikto ALI schematinis pavaizdavimas,
- FIG. 1B - taktinių ir kelių pagrindinių ALI valdymo signalų, pavaizduotų Fig. 1A, diagramos,
- FIG. 2 - ALI realizavimo pagal išradimą blokinė schema,
- FIG. 3 - schematinis bitų lauko padalinimas,
- FIG. 4A - grafinis sveiko dydžio kodo pavaizdavimas,
- FIG. 4B - grafinis slankaus kabelio dydžio kodo pavaizdavimas,
- FIG. 4C - schematinis aritmetinių operacijų, atliekamų su 7 bitų magistrale, pavaizdavimas,
- FIG. 5A - 5D - parodytas duomenų sąrašas, gaunamas atitinkamai su FIG. 4,
- FIG. 6 - ALI realizavimo pagal išradimo aprašymą schema,
- FIG. 7A - 7C - skaitmeninio ALI pagal išradimo aprašymą įėjimo/išėjimo buferio realizacija, esant skirtingoms taktinio signalo fazėms,

FIG. 8 - operando įtaiso, pavaizduoto schemeje FIG. 6, išėjimo signalų kombinacijos kartu su mneumonika, naudojama elementų, esančių operando įtaise, valdymui ir skirtingų išėjimo kombinacijų formavimui,

FIG. 9 - tikslumo dekodierio, pavaizduoto schemeje FIG. 6, išėjimai, kai į vieną iš įėjimo magistralių paduotas žodis, kuris pateikiamas kaip pavyzdys.

- FIG. 10 - binarinės mantinės bitų kombinacijų, duotų kaip pavyzdys, skirtigos interpretacijos,
- FIG. 11 ir 12 - diagramos, papildančios dviejų slankaus kabelio skaičių sudėties pavyzdžio aprašymą,
- FIG. 13 - tikslumo dekodierio, pavaizduoto schemeje FIG. 6, blokinė schema,
- FIG. 14A - pilno žodžio sudėties įtaiso (sumatoriaus), pavaizduoto schemeje FIG. 6, blokinė schema,
- FIG. 14B - sumatoriaus FIG. 14A tikslumo išpjovos blokinė schema,
- FIG. 14C - tikslumo išpjovos FIG. 14B bitų išpjovos blokinė schema,
- FIG. 15 - tikslumo išpjovos inkrementoriuje, pavaizduotame FIG. 6, blokinė schema,
- FIG. 16 - operandų įtaiso, pavaizduoto Fig. 6, selektoriaus bazinės bitų išpjovos realizavimo schema.

DETALUS IŠRADIMO APRAŠYMAS

FIG. 1

Skaitmeninis aritmetinis loginis įtaisas ALI, schematiškai pavaizduotas FIG. 1A, sudarytas iš valdymo 1a ir aritmetinės matricos 2a įtaisų, atliekančių aritmetines operacijas su skirtingais įėjimais v0, v1, v2 ir v3, kiekvienas kuris yra

magistralė, turinti eilę (M) perdavimo linijų, perduodančių M bitų dydžius. Pavyzdžiui M gali būti lygus 32. ALĮ, dažniausiai, yra centrinio procesoriaus įtaiso CPI dalis ir yra išoriškai valdomas valdymo įtaiso VĮ. ALĮ keičiasi informacija su išoriniu valdymo įtaisu CĮ šiais signalais: taktinio dažnio signalu - įėjimas C_{ALU} , užimtumo signalu - išėjimas A_{BUSY} , valdymo signalu - įėjimas A_{CNTL} , būsenos signalu - išėjimas A_{STATE} , ir dekodutos komandos signalu - išėjimas A_{INS} .

Įėj/išėjimo magistralės v0, v1, v2 ir v3 gali būti sujungtos su skirtingais registrais, vadinamais tikslo registrais arba G-registrais, esančiais struktūriniame aritmetiniame įtaise (neparodytas), žemiau vadinamame pagrindiniu procesoriumi, kuriame apdorojamos komandos. Vienok, signalai į įėj/išėjimo magistralės gali būti paduodami ir iš kitų, geriausia su sumažinto komandų skaičiaus tipo, įrenginių, dirbančių kartu su aritmetiniu loginiu įtaisu. Galimas ir kito tipo skaičiavimo įrenginių prijungimas. Taip pat, įėj/išėjimo magistralių v0, v1, v2 ir v3 skaičius, kuris, kaip, parodyta FIG. 1A, yra keturi, gali būti laisvai užsiduodamas, tačiau ALĮ sudėtingumas didėja proporcingai įėj/išėjimo v0, v1, v2, v3 skaičiui.

Įėj/išėjimo magistralės v0, v1, v2, v3 perduoda žodžių sąrašą. Žodyje yra arba skaitmeninis dydis arba komanda. Visos magistralės negali būti naudojamos visą laiką. Magistralėje esantis nenaudojamas žodis turi būti ignoruojamas. Įėjimas ir/ arba išėjimo sąrašas yra didžiausio ilgio, t.y. sudarytas iš maksimalaus žodžių skaičiaus, pavyzdžiui iš keturių, t.y. vienas sąrašo elementas priskiriamas vienai magistralei. Skaitmeninis dydis - žodis, kuris perduodamas keletu linijų. Komanda magistralėje taip pat yra žodis. Įėjimo sąrašas, paduotas į magistralės v0, v1, v2, v3, naudojamas skaičiavimų vykdymui ir valdymui, t.y. sąraše pateikiama informacija apie skaičiavimų, kurie bus atliekami, tipą ir dydžio, su kuriuo bus vykdomi skaičiavimai, požymius, ir valdymo įtaisas la veda šią informaciją iš sąrašo, tol kol matricoje 2a tikrinamos įvairios komponentės. Skaičiavimai atliekami perrašant įėjimo sąrašo turinį į bitų ar bitų grupę skirtingais būdais, t.y. į bitus, esančius toje pačioje žodžio vietoje, įrašoma atskirai, o į bitų grupės paduodamos į žodį „išpjovų būdu, ir rezultatas pateikiamas kaip išėjimo sąrašas. Reikia pažymėti, kad galima atlikti perrašymus vidinėse grandinėse keletą kartų iki galutinio išėjimo sąrašas nebus suformuotas išėjime. Vidinio perrašinėjimo metu momentinis išėjimo sąrašas išvedinėjamas į išėjimą, bet yra įterpiamas tiksliai kaip momentinis įėjimo sąrašas.

Į sąrašą leidžiama įtraukti kartu su apdorojamais dydžiais ir panaudojimo funkciją arba kito tipo sąrašą, pavyzdžiui, sudarytą tik iš, mažiausiai, vieno sveiko arba slankaus kablelio dydžio. Kai sąraše yra panaudojimo funkcija, pirmajame sąrašo elemente, paduodamame į įėj/išėjimą v0, yra komandos kodas, t.y. informacija apie būsimo apdorojimo tipą, o laisvi sąrašo elementai yra komandos argumentai. Detalesnis aprašymas bus pateiktas žemiau.

Komanda yra vykdoma, ją perrašant vienu arba keliais žingsniais, taip, kad galų gale gauti rezultatą. Mikrokomandos ir būsenos signalai naudojami ALĮ viduje. Juos naudoja atskiros valdymo įtaisos la, esantis ALĮ viduje, matricos 2a valdymui. Kai perrašymo metu, iškart, gaunamas rezultatas, šis dydis atspindimas išėjimo sąraše. Kai perrašymas atliekamas žingsniais, sąrašas perrašinėjamas, išsaugant tą patį komandos kodą, arba paduodant naują arba mažai pakeistą komandos kodą, po kurio paprastai seka vienas dydis ar daugiau.

Reikia pažymėti, kad geriausias perrašymo operacijos keliais ciklais atlikimo būdas yra padavinėti komandos kodą nepakeistą visos operacijos metu. Šiuo atveju valdymo įrenginio schema yra paprastesnė. Vienok, kai kuriais atvejais patogiau, kai daugiacyklės operacijos pradžioje buvęs komandos kodas lėtai keičiamas likusiuose cikluose arba panašiai.

Grįžtant prie FIG. 1A, taktinio dažnio signalas C_{ALU} naudojamas sinchronizuoti ALĮ darbui ir pats aritmetinis loginis įtaisas (ALĮ) signalu išėjime A_{BUSY} informuoja apie pasiruošimą priimti taktinius impulsus.

Magistralių v0, v1, v2, v3 įėj/išėjimai, perduodantys komandas arba skaitmeninį turinį, yra prijungti prie pagrindinio registro ląstelių tikslo registre. Skaitmeninį turinį geriausia pateikti skirtingai užkoduotą, kai jame yra slankaus kablelio arba sveiko tipų dydžiai. Kiekviena tikslo registro ląstelė susideda iš tiek bitų, kiek yra skaitmeninio dydžio įėj/išėjimų (t.y. 32 bitai parodytame pavyzdyje), bet naudojama keletas papildomų bitų, pavyzdžiui 6, į kuriuos įrašoma antraštės. Šios antraštės neįtraukiamos į informaciją, perduodamą į įėj/išėjimo magistralės, šiame išradimo aprašyme. Vietoj to, antraštės turinys, apeinant ALĮ valdymo įtaisą la, perduodamas į išorinį valdymo įtaisą VĮ.

Vienok, antraštės įjungimas į informaciją, paduodamą į įėj/išėjimus, tiesiogiai sujungtus su valdymo įtaisu arba su ALĮ valdymu, neįeina į išradimą, bet tai būtų geriausias ALĮ, pateikto išradime, panaudojimo su įprasto tipo procesoriumi, valdymo būdas.

Valdymo įėjimas A_{CNTL} naudojamas skaitmeninių operacijų ALĮ valdymui. Operacijos gali užtrukti keletą taktinių ciklų.

Signalas į išėjimą A_{STATE} grąžinamas po kiekvienos operacijos valdymo įrenginiui CĮ. Komandos kodas išėjime A_{INS} yra kompaktizuotas procesoriaus tikslo ląstelės, prie kurios prijungta magistralė v0, komandos kodas.

FIG. 1B

Pagrindinis procesoriaus taktinio dažnio signalas C10 yra išduodamas valdymo įrenginio VĮ. Šio signalo trukmė, pavyzdžiui, gali būti 10 ns. ALĮ operacijas vykdo vieno arba kelių šio taktinio dažnio periodų metu. Pagrindinis procesorius, kuris yra sujungtas su ALĮ gali būti valdomas mažesniu nei ALĮ taktiniu dažnio signalu, pavyzdžiui du kartus, t.y. taktinio signalo C20 pasikartojimo periodas yra 20 ns, kai C10 yra 10 ns. Vienok, juos galima valdyti ir aukštesnio dažnio taktiniu signalu. Pagrindinis valdymo signalas C_{ALU} įtaissams turintiems trumpesnį taktinio dažnio signalo periodą (pavyzdžiui, ALĮ šiame realizacijos aprašyme arba kitur pagrindiniame procesoriuje) yra pailginamas komandos pabaigoje, tam kad sinchronizuotis su signalu C 20, turinčiu ilgesnį pasikartojimo periodą. ALĮ formuoja aukšto lygio signalą A_{BUSY} tiek ilgai, kiek reikia taktinio dažnio signalo C 10 ciklų komandai atlikti. Signalas C ALĮ pailginamas po to kai A_{BUSY} pereina į žemą lygį (žiūrėti FIG. 1B).

FIG. 2

Išradime aprašyto ALĮ realizaciję parodytoje FIG. 2. pateikta ALĮ valdymo grandinė 10, kaip centrinis elementas, atliekantis įėjimo signalų pakeitimus ir apjungimus atinkamai kitoms grandinėms, kad suformuoti išėjimo sąrašo elementus. ALĮ valdymo grandinėje 10 yra ir loginių elementų programuojama matrica. Kiekviena įėj/išėjimo magistralė v0, v1, v2, v3, būdama daugialaidė magistralė, sudaryta, pavyzdžiui, iš 32 linijų, yra sujungta su buferinės grandinės 11 įėjimais. Taktiniu signalu per taktavimo įėjimą yra valdoma, buferinė grandinė 11 perduoda įėjimo signalus į vidines įėjimo magistrales v0i, v1i, v2i, v3i. Išėjimo buferis 16 per taktinį įėjimą, taktuojamas invertuotu taktinio dažnio signalu, perduoda informaciją iš vidinių magistralių v0out, v1out, v2out, v3out į įėj/išėjimo magistrales v0, v1, v2, v3. Buferiai 11 ir 16, kaip parodyta FIG.2, yra du atskiri buferiai, tačiau geriausia naudoti kombinuotą įėjimo/išėjimo buferinę grandinę.

Eilė skaičiavimo grandinių 12A - 12N, kur N grandinių skaičius, yra sujungtos su mažiausiai dviem vidinės magistralių v0i, v1i, v2i, v3i įėjimais. Kiekviena grandinė atlieka atskirus paprastus skaičiavimus, tokius kaip sudėtis ir atimtis skirtingose dalyse, su informacija paduodama iš vidinių įėjimo magistralių v0i, v1i, v2i, v3i. Skaičiavimo grandinės 12A - 12N yra su atmintimi, kadangi jos kiekvienam galimam paprastam skaičiavimui, reikalingam panaudojime, kuriame ALĮ pagal išradimą yra dalis, nustatomos vieną kartą. ALĮ valdymo grandinė 10 tada gali tiesiogiai išrinkti vienos, tinkamos skaičiavimo grandinės kiekvienam dalinės operacijos tipui, išėjimą tolimesnio apdorojimo tikslams. Kiekviena skaičiavimo grandinė yra nesudėtinga ir vartoja mažai energijos. Taigi, nėra jokių kliūčių naudoti kelias, lygiagrečiai skaičiuojančias grandines, vienu metu nuskaitant tik vienos išėjimą. Skaičiavimo grandinių išėjimai yra prijungti prie operando įtaisų OP 1 ir OP 2, kurie bus aprašyti žemiau.

Įtaisų 12A - 12N signalai, eilei atvejų, gali būti naudojami valdymui. Kiekviena skaičiavimo grandinė 12A - 12N turi antrąjį įėjimą, į kurį paduodamas perkėlimo signalas, naudojamas indikuoti perpildymą. Šie antrieji įėjimai yra prijungti prie atskirų ALĮ valdymo grandinės 10 įėjimų. Įtaisai 12A - 12N valdomi grandinės 10 ir kitų ALĮ grandinių, duodančių informaciją apie į jų įėjimus pateikto žodžio tipą, t.y., pavyzdžiui, ar žodis atspindi skaičių ar kita, ar skaičius yra sveikas ar slankaus kablelio dydis ir, jeigu tai slankaus kablelio dydis, informaciją apie šio skaičiaus savybes.

Eilė monitoringo grandinių 13A - 13M, kur M grandinių skaičius, yra prijungtos mažiausiai prie vienos iš įėjimo magistralių v0i, v1i, v2i, v3i. Monitoringo grandinės gali, pavyzdžiui, parodyti informacijos magistralėse reikšmingumą arba sulygtinti magistralėse esančią informaciją. Keletas monitoringo grandinių gali gauti tą pačią valdymo informaciją kaip ir grandinės 12A - 12B, kaip tai parodyta FIG. 2 grandinei 13A. Monitoringo grandinių 13A - 13M išėjimai yra prijungti prie ALĮ valdymo grandinės 10 įėjimų.

Trys iš keturių požymius išskiriančios grandinės 14A - 14D yra prijungtos prie mažiausiai šių magistralių v1i, v2i, v3i. Jos nuveda informaciją nuo dalies magistralių, duodamos signalus atsižvelgiant į likusios informacijos magistralėse savybes. Požymius išskirianti grandinė 14D prijungta prie magistralės v3i nuveda, mažiausiai, mažiausiai reikšmingą dvejetainį skaičių šioje magistralėje daugybos metu, kaip bus parodyta žemiau. Požymius išskirianti grandinė 14A, taip pat, gali būti prijungta prie magistralės v0i, kad nuvestų išanksto užduotą informaciją, tuo atveju, kai signalai išorinėje valdymo magistralėje EXT į ALĮ valdymo įrenginį rodo, kad informacija magistralėje v0i atspindi skaičių.

Pavyzdžiui, jeigu informacija magistralėje yra slankaus kablelio dydis, tai požymius išskirianti grandinė, prijungta prie magistralių, duos informaciją atspindinčią dydžio ženklą, eksponentės ženklą ir kelis mantinės bitus, pavyzdžiui reikšmingiausius. Reikia pabrėžti, kad požymis, grandinės 14D perduodamas iš v3i magistralės į vieną ALĮ valdymo grandinės 10 įėjimą, neturi būti toks pat kaip iš magistralių v1i ir v2i (tai mažiau sudėtinga). Tie patys motyvai tinka ir požymius v0i magistralėje išskiriančiai grandinei 14A.

ALĮ valdymo grandinė 10 išduoda išėjimo signalą A_{STATE} į išorinį valdymo įrenginį VĮ (žiūrėti FIG. 1) priklausomai nuo įėjimo signalų ateinančių iš įtaisų 12A - 12N ir 13A - 13M.

Komandos grandinė 15 naudojama visos informacijos iš magistralės v0i tikrinimui tam, kad duoti atitinkamų operacijų, kurias turi atlikti ALĮ valdymo grandinė 10, dekoduoti komandas.

Kaip parodyta FIG. 2 informacija iš magistralės v0i gali būti paduota tiesiogiai į išėjimo magistralę v0out, sujungtą su išėjimo buferiu 16, geriausia kai magistralė prijungiama prie kombinuoto įėjimo/išėjimo buferio 11, 16, taktuojamu invertuotu taktiniu signalu, kaip buvo minėta aukščiau. Yra patogu, kad ši magistralė būtų komandų sąrašo elementas ir šis komandų sąrašo elementas nebūtų keičiamas. Kitos išėjimo magistralės v1out, v2out, v3out taip pat yra sujungtos su buferiu 16, ir informacija, esanti visose išėjimo magistralėse, išėjimui į įėj/išėjimo magistrales v0, v1, v2, v3 taktuojama vienu metu. ALĮ valdymo grandinė 10 paduoda valdymo signalus c1, c2, c3, c4, c5 į kitas ALĮ grandines.

Vienok, galima, mažiausiai dalį v0i magistralės sujungti su ALĮ valdymo grandine 10, tai iššauktų keletą pakeitimų priklausomai nuo vykdomo ciklo daugiacyklėje operacijoje, ir paduoti pakeistą sąrašo elementą į magistralę v0out (neparodyta).

Du operando įtaisai OP1 ir OP2, iš esmės vienodi, aptarnauja išėjimus v1out ir v2out, atitinkamai. Nebūtina, tačiau praktiška turėti du operando įtaisy kaip parodyta realizacijoje FIG. 2. Skirtingas operandų įtaisų skaičius, nuo vieno iki kelių, gali būti panaudotas įvairiems ALĮ, pagaminto pagal išradimą, pritaikymams. Tikrai vienas operando įtaisy reikalingas sudėčiai ir atimčiai atlikti, kadangi šios operacijos vykdomos vienoje iš 12A - 12N grandinių ir rezultatas yra paduodamas į vieną iš operando įtaiso įėjimą ir gali būti paduodami iš operando įtaiso išėjimo į ALĮ valdymo grandinę 10. Keletas operandų įtaisų gali būti naudojami tada, kai keletą skirtingų sudėties ir atimties operacijų reikia atlikti vienu metu. Turi būti pažymėta,

kad kiekvienas iš operando įtaisų sujungtas su magistrale vouti, kur i užduodamas pasirinktam skaičiui, ir kad, įėj/išėjimų, įėjimo ir išėjimo magistralių skaičius priklauso nuo operandų įtaisų skaičiaus, panaudotų ALĮ.

Kito tipo operando įtaisas OP3 naudojamas daugybos ir dalybos operacijose ir perduoda išėjimo signalus į išėjimo magistralę v3out. Daugybos ir dalybos operacijos atliekamos keliais cikliniais žingsniais. Operando įtaiso OP3 išėjimas yra v3out. Taip pat pažymėsime, kad gali būti skirtingas nei vienas operando įtaisų OP3 skaičius.

Apie slankaus kablelio aritmetiką, kaip buvo minėta, detaliau kalbėsime žemiau. Eksponentės ilgi pasirenkamas, mažiausiai, žingsniais, mažiausiai, vienoje iš išradimo realizacijų. Be to, kiekvienam operando įtaisui OP1 ir OP2 priskiriami atitinkami tikslumo dekodieriai PD1 ir PD2. Tikslumo dekodieris paduoda į atitinkamą operando įtaisą valdymo signalą, įvertinantį eksponentinės dalies ilgį. Tai gali būti gauta iš dalinio kodo lauko binariniam žodyje, o kaip tai atliekama bus parodyta žemiau. Tikslumo dekodieris, kaip buvo minėta, perduoda valdymo signalus ir į kitas ALĮ grandines.

Informacija į ALĮ, kurioje įvertinama ar dydis yra sveikas ar slankaus kablelio, tinkamiausiai paduodama į ALĮ valdymo grandinės 10 valdymo įėjimą EXT išoriniu valdymo įrenginiu VĮ (žiūrėti FIG. 1A), tačiau gali būti nustatoma iš magistralių v0i - v3i.

Kaip seka iš aukščiau, atliekant aritmetinius skaičiavimus tikslinga, kad įėjimo magistralėmis perduodama informacija būtų skirtinga ir apibrėžta. Taigi, įėjimo magistralės, atitinkama, apibrėžto turinio informacija, aprūpinamos tiesiogiai iš įtaisų, tokių kaip, tikslo registras, minėtas aukščiau, kuris yra pageidautinas. Kitaip, ALĮ turės pats keisti informaciją įėjimo magistralėse, būdamas susijęs vidiniame cikle, ir tada naudodamasis informacija, esančia magistralėje v0i, galės įvertinti informaciją kitose magistralėse. Magistralėje v0i gali būti duomenų laukas, kuriame yra sąrašo, paduodamo į įėj/išėjimo magistralės v0, v1, v2, v3, informacijos tipo įvertinimas. Be to, ši informacija gali būti paduodama iš išrinio valdymo įrenginio VĮ.

Kaip seka iš aukščiau paminėto, ALĮ, realizuotam pagal išradimą, išorinė atmintis nereikalinga. Tai duoda, kad kiekvieno ciklo rezultatas daugiacyklyje operacijoje yra įtakotas tik tai prieš tai buvusio ciklo išėjimo rezultatu, kuris yra įėjimas, pritaikytas tolimesniam apdorojimui.

Kaip buvo minėta aukščiau, vienas iš tikslų, priskiriamų ALĮ pagal išradimą, yra skirtingo padalinimo tarp eksponentės ir mantinės dalių slankaus kablelio dydžių atspindėjimo realizavimas. Tada, iš vienos pusės, galima turėti daug bitų mantinėje mažiems skaičiams atspindėti, kas duoda pranašumą esant dideliui tikslumui. Iš kitos pusės, galima apdoroti ir labai didelius skaičius.

SVEIKU SKAIČIŲ DAUGYBOS PAVYZDYS

Tam, kad trumpai ir suprantamai paaiškinti, pateikto išradime, ALĮ veikimą čia duodamas dviejų sveikų skaičių daugybos pavyzdys. Toliau bus duodamas pavyzdys, paaiškinantis ALĮ veikimą, kai skaičiavimai vykdomi su slankaus kablelio dydžiais.

Sveikų skaičių daugyba gali būti atliekama sekančiu būdu. Pirmojo ciklo metu du dydžiai, kuriuos reikia sudauginti, paduodami į įėjimo magistralę v3. Magistralėje v0 yra informacija „0“. Vienas, pirmojo tipo operando įtaisas, pavyzdžiui OP1, valdomas, kad perduotą informaciją tiesiai į v2 išėjimą. Kitas operando įtaisas, OP2, valdomas taip, kad pastumtą informaciją, esančią v2 vienu žingsniu į kairę ir atspindėti pastumtą rezultatą savo išėjime. Sekančio ciklo metu, pirmasis operando įtaisas, OP1, yra valdomas perduoti informaciją iš vieno iš įėjimų, sujungto su viena iš grandinių 12A - 12N, atliekančių dydžių, esančių magistralėse v1i ir v2i sudėti įėjimą. Kitas operando įtaisas, OP2, valdomas taip, kad paimtą informaciją iš magistralės v2i ir pastumtą ją vienu žingsniu į kairę, o po to išvestų į išėjimą.

Po kiekvieno ciklo, pirmasis operando įtaisas, OP1, valdomas taip, kad perduotą, grandinės, atliekančios informacijos sudėti, išėjimą į magistralės v1 ir v2, jeigu paskutinis informacijos binarinis dydis magistralėje v3i yra „1“ arba, kitu atveju, nuskaito informaciją iš magistralės v1i. Ši informacija į ALĮ valdymo grandinę 10 paduodama iš grandinės 14D išėjimo. Informacija, apie dydį magistralėje v3i, paduodama į operando įtaisą OP3, kuris valdomas taip, kad pastumtą jį vienu žingsniu į dešinę ir, kad mažiausiai reikšmingas binarinis skaičius būtų išstumtas. Operando grandinės OP1 ir OP2, mažiausiai, turi turėti sąvybę formuoti grįžimo signalą (neparodytas FIG. 2), paduodamą atgal į ALĮ valdymo grandinę 10, indikuojantį, kad vidinis postūmis davė perkėlimo signalą. Kiekvienas ciklas atliekamas dviem žingsniais, pirmasis - pirmąjį taktinio signalo periodo dalį, kitas - antrosios dalies metu.

FIG. 3

Kaip schematiškai parodyta FIG. 3, kiekvieno magistrale paduodamo bitų paterno, atspindinčio skaitmeninį dydį, reikšmingiausiuose bituose išskiriamas trumpas bitų laukas, kuriame, t.y. bitų sekoja, atspindimas dydžio ženklas, eksponentės ženklas ir kodas, įvertinantis skaitmeninio dydžio padalinimo į eksponentės ir mantinės dalis vietą (žiūrėti dešiniame stulpelyje FIG. 3). Reikia pažymėti, kad kodo dalies ilgis bitų lauke gali kisti ir, kad padalinimo vieta tarp kodo lauko ir eksponentės lauko yra apibrėžta kodo lauku, kaip pavaizduota kairiajame stulpelyje FIG.3.

Trys dešimtinių skaičių atspindėjimo būdai bus aptarti toliau:

- | | |
|--------------------------------|---|
| I Dvejatinė forma | Skaitmeninė sistema, kurios pagrindas 2. |
| II Sveikas H -dydis | Binarinis kodas sveikų dydžių atspindėjimui. |
| III Slankaus kablelio H -dydis | Binarinis kodas slankaus kablelio dydžių atspindėjimui. |

Kitas tikslas, kuris turi būti įgyvendintas ALĮ pagal išradimą, tai atspindėti slankaus kabelio tikslumu duotą skaičių ta pačia tvarka kaip yra interpretuojamas toks pats sveikas skaičius. Todėl, toks pat bitų paternas visiškai skirtingai interpretuojamas kaip slankaus kabelio dydis ir sveikas dydis. Vienok, šis tikslas gali būti įgyvendintas turint pirmą bitų sekos interpretacijos formulę, kai atspindimi slankaus kabelio dydžiai, ir antrą bitų sekos interpretacijos formulę, kai atspindimi sveiki dydžiai. Taigi nei vienas iš atspindėjimų neturi būti dvejetainis, vietoj to, mažiausiai vienas iš jų, o geriausiai abu, turi būti užkoduoti. Bitų paterno transformavimas atliekamas automatiškai ir aparatūra mašinoje atlieka skaičiavimus su transformuotais paternais tik besąlygiškai laikantis duotų taisyklių, t.y. nežinant kaip interpretacija buvo atlikta.

Pagal geriausią išradimo realizaciją, atliktas žingsnis, gauti sutankintą slankaus kabelio dydžių atspindėjimą, t.y. neturėtų būti dviejų paternų, kurie būtų atspindimi kaip toks pat skaičius. Terminas „artimas atspindėjimas“ reiškia, kad kiekvieną dešimtainį skaičių atitinka vienintelė binarinė kombinacija.

FIG. 4A. SVEIKŲ DYDŽIŲ ATSPINDĖJIMAS

FIG. 4A parodytas sveiko dydžio, vadinamo sveiku H - dydžiu, kodo, pagal išradimą, grafinis pavaizdavimas. Binarinis atspindėjimas duodamas išilgai horizontalios ašies, o H - dydžiai išilgai vertikaliosios ašies. Reikšmingiausias bitas atspindi dydžio ženklą, t.y. ar dydis yra teigiamas ar neigiamas. Kaip matome iš FIG. 4A, H - dydis yra labiausiai neigiamas, kai binariniame atspindėjime yra visi „0“, nulis yra ir binarinio atspindėjimo viduryje, t.y., kai reikšmingiausias bitas keičiasi iš „0“ į „1“, ir yra labiausiai teigiamas dydis, kai binariniame atspindėjime visi bitai yra „1“. Tokiu būdu, binarinės kombinacijos dydis H - kode didėja tuo pačiu būdu kaip tai vyksta dvejetainėje skaičiavimo sistemoje. Kai žodžiu dvejetainėje kombinacijoje yra užpildytas „1“, tai reiškia jo maksimalią vertę. Sveiko dydžio kodas atspindimas tiesia linija.

FIG. 4B. SLANKAUS KABELIO DYDŽIO ATSPINDĖJIMAS

FIG. 4B parodytas slankaus kabelio dydžio, vadinamo slankaus kabelio H - dydžiu, kodo, pagal išradimą, grafinis pavaizdavimas. Binarinis atspindėjimas duodamas išilgai horizontalios ašies, o H - dydžiai, kurie čia nubrėžti per tašką kur H - dydis keičia ženklą, išilgai vertikaliosios ašies. Reikšmingiausias bitas čia taip pat atspindi dydžio ženklą. Slankaus kabelio dydžių atspindėjimas užbaigia universalią atspindėjimo sistemą. Priklausmai nuo tikslumo, dydžio reikšmė gali būti skirtinga. Tokiu būdu, labai maži ir labai dideli dydžiai galėtų būti atspindinami. Slankaus kabelio H - dydžių skalė yra logaritminė. Logaritminis H - dydis gali būti supaprastintas naudojant tiesinę aproksimaciją tarp kiekvieno iš dviejų faktorių. Rezultatas yra tipinis slankaus kabelio kodas. Kaip parodyta FIG. 4B, H - dydžių, kintančių nuo -1 iki +1, binarinis atspindėjimas gaunamas binarinio atspindėjimo lauko viduryje ir užima pusę lauko dydžio.

Logaritminis dydis gali būti panaudotas logaritminėje funkcijoje vieną ar daugiau kartų. Taigi, įmanoma atspindėti labai mažus ir labai didelius dydžius. Vienok, esant labai didelėms reikšmėms tikslumas mažėja, tačiau, mažų reikšmių dydžiams, tikslumas gali išlikti pastovus arba nežymiai didėti.

Kadangi, du bitų paternai dviejuose atspindėjimuose gauti tuo pačiu būdu, tai sveikų dydžių komparatoriai taip pat gali būti panaudoti slankaus kabelio dydžių atspindėjimui. Tai reiškia, kad atskiras komparatorius slankaus kabelio dydžiams gali būti panaikintas. Arba, kitaip sakant, sveiko dydžio komparatorius, atliekantis palyginimo operaciją per vieną ciklą, gali būti panaudotas slankaus kabelio dydžiams palyginti, o tai yra didelis privalumas. Įprastas slankaus kabelio komparatorius naudoja daug ciklų palyginimo operacijai atlikti.

Įprastas aritmetinis loginis įtaisas keletą bitų paternų interpretuoja kaip tokius pat skaičius ir tik normalizacijos dėka išrenkamas atitinkantis paternas. Išradime pateiktame ALĮ normalizacija nereikalinga, nors jau skaičiavimo metu. Dydžiai gali būti normalizuojami prieš arba po skaičiavimų. Kadangi, sutankintas ir slankaus kabelio dydžių atspindėjimai yra panašūs, tai slankaus kabelio dydžio skaičiavimai, tokie kaip palyginimas atliekami vieno ciklo metu. Kitas sutankinto atspindėjimo privalumas yra tai, kad gaunamas maksimalus slankaus kabelio dydžių skaičius, t.y. nenaudojami ir atmeti dydžiai nepadaudami į dydžių seką.

Kelių bitų paternų interpretavimo kaip to paties skaičius, tradiciniuose ALĮ, priežastimi yra tai, kad eksponentės ir mantisės dalys, norint gauti skaičių, sudauginimos. Pavyzdžiui dydis $1,0$ gali būti išreikštas kaip $2^{1/2}$, t.y. eksponentės bitų paternas - 01, o mantisės bitų paternas 100, arba $2^{2/4}$, t.y. eksponentės bitų paternas - 10, o mantisės bitų paternas - 010, eksponentės ir mantisės bitų paternai išdėstyti vienas paskui kitą.

Pagal išradimo aprašymą, sutankintas atspindėjimas sukuriamas nustatant „1“ prieš mantisės bitų paterną teigiamais dydžiams ir „0“ - neigiamais. Įprastai mantisė charakterizuojama kaip dydis, kintantis nuo -1 iki +1. Pagal išradimą binariniais dydžiais koduota mantisė charakterizuojama kaip dydis kintantis diapazonuose (nuo $1,0$ iki $2,0$) arba (nuo $-2,0$ iki -1) priklausomai nuo ženklo.

Sveikų dydžių paternai yra tiksliai apibrėžti. Slankaus kabelio dydžio atspindėjimas turėtų turėti tokią pat nuoseklią tvarką kaip ir sveikų interpretacijoje, o tai yra labai didelis apribojimas.

FIG. 4C. 5A - 5D. TOLIMESNĖ INFORMACIJA APIE SLANKAUS KABELIO DYDŽIŲ ATSPINDĖJIMA

Optimaliausias šios problemos sprendimas schematiškai pavaizduotas FIG. 4C. Paveikslai FIG. 5A - 5D demonstruoja duomenų žodžių sąrašą, pateikiamą pagal principus parodytus FIG. 4C. Stulpelis toliausiai dešinėje paveiksluose FIG. 5A - 5D parodo dešimtainį slankaus kabelio dydį, kuriam atitinka binarinė kombinacija, esanti kairiajame kraštiniame

stulpelyje. FIG. 4C parodytas grafinis slankaus kabelio kodo pavaizdavimas, binarinis atspindėjimas parodytas kairėje vertikalių linijų LIN pusėje, slankaus kabelio H - dydžiai dešinėje LIN pusėje. Linija LIN nubrėžta kaip ašis, rodanti didesniųjų dydžių kryptimi. Tai tinka abiem, tiek binariniam tiek slankaus kabelio dydžių atspindėjimams.

Binarinis atspindėjimas ir H - dydžiai yra pavaizduoti iš priešingose linijos LIN pusėse. Artimiausios linijai LIN yra ašių, parodytų FIG. 4B, iliustracijos. H - dydžio ašis dešinėje ir binarinio atspindėjimo ašis kairėje.

Pirmasis laukas šalia ašių pavaizdavimo, sh - dešinėje ir s - kairėje, atspindi reikšmingiausią bitą. Jis yra „0“ - neigiamais ir „1“ - teigiamais dydžiams.

Sekantis po pirmojo, laukas seh ir se, atitinkamai, atspindi reikšmingiausius bitus.

Trečiajame lauke parodytas vienas bitas, turintis mažesnę reikšmingumą nei sekantis, reikšmingiausias bitas; vienas bitas, kai laukas mažas, kaip iliustruojamame atvejuje tik su septyniais bitais, ir daugiau, kai binarinis laukas didelis, kaip bus paaiškinta atvejui su 32 bitais.

Pavyzdyje, parodytame FIG. 4C, 5A - 5D, demonstruojamas septynių bitų žodžio, vietoj 32 bitų žodžio, kuris dažniau naudojamas praktikoje, kodavimas. Binarinis atspindėjimas centrinės linijos kairėje paveikslė FIG. 4C interpretuojamas kaip H - dydžiai, kintantys nuo -1024 iki +768. Turi būti pažymėta, kad H - dydžiai -1 ir +1 yra išdėstyti tarp dydžio 0 ir maksimalių dydžių -1024 ir 768, atitinkamai. Pirmasis binarinis bitas s ir sh, atitinkamai abiem atspindėjimams, t.y. binariniam ir H - dydžio, yra dydžio ženklas, kuris yra teigiamas arba neigiamas.

Sekantis bitas yra eksponentės ženklo bitas, t.y. dydis kurio absoliuti vertė yra didesnė arba mažesnė už vienetą. H - dydžio eksponentės ženklo bitas seh yra teigiamas tarp -1024 ir -1, neigiamas tarp -1 ir +1, ir vėl teigiamas nuo +1 iki +768. Kaip matyti iš FIG. 4C kairiosios dalies, binariniam atspindėjimui eksponentės ženklo bitas se yra neigiamas pirmajame, teigiamas antrajame, neigiamas trečiajame ir teigiamas ketvirtajame binarinės sekos ketvirčiuose. Bitas seh gali būti aprašytas kaip seh=se, kai s=1, ir seh=1-se, kitais atvejais.

Taigi, binariniame atspindėjime reikšmingiausias bitas, atspindintis simbolio ženklą, kinta mažiausiai, sekantis mažiau reikšmingas bitas, atspindintis eksponentės ženklą kinta du kartus greičiau, t.y. du kartus greičiau nei keičiasi dydžio ženklas. Sekantis, trečiasis reikšmingumo mažėjimo kryptimi stovintis bitas, kinta greičiau nei eksponentės ženklo bitas ir vadinamas kodo bitu c. Parodytame FIG. 4C, 5A - 5D pavyzdyje kodą sudaro tik vienas bitas. Kituose pavyzdžiuose, kodo lauką gali sudaryti ir daugiau bitų. Kodas valdo eksponentės ilgį ir nustatomas prieš einančio ženklo bito atžvilgiu.

Gautas slankaus kabelio H - dydis, t.y. slankaus kabelio sistema, naudoja kodo lauką skirtingai. Kaip matyti iš eh dalies FIG. 4C, kuri tiesiogiai parodo H - dydžio eksponentę, dydis eh keičiasi nuo didelės teigiamos reikšmės iki 0, kur H - dydis yra -1, nuo čia iki didelių neigiamų reikšmių, kur H - dydis yra 0, nuo čia iki 0, kur H - dydis yra +1, ir nuo čia iki didelių teigiamų reikšmių, kur H - dydis yra 768. Sekcija eh abs rodo absoliučias H dydžio eksponentės reikšmes ir yra pavaizduota kaip dvi vienodos, taške kai H - dydis lygus nuliui, tarpusavyje sujungtos, turinčios didelės teigiamos reikšmės galuose, kreivės. H - dydžių kodo laukas ch sukurtas atspindėti eh abs ir, kaip matyti iš FIG. 4C turi tokią pat formą (trūki linija). Ištinę linija pavaizduota ch forma kai kodo laukui naudojamas tik vienas bitas. Tokiu būdu, kai kodo laukas sudaro vienas bitas, kodas aprašomas ch=c, kai se=1, ir ch=1-c, kitoms reikšmėms.

Iki čia buvo nesunku sekti sąrašo, parodyto FIG. 5A - 5D, paaiškinimą, kadangi kiekvieną iš bitų laukų s, sh, se, seh, c ir ch sudarė tik vienas bitas ir palyginimai tarp atspindėjimų buvo akivaizdūs. Pirmasis stulpelis FIG. 5A - 5D yra nuoseklus binarinis sąrašas nuo 0000000 iki 1111111, kurio reikšmės dešimtainėje sistemoje parodytos gretimame stulpelyje. Dešiniau rodyklės stulpelio yra pavaizduoti transformuoti atspindėjimai skaičiai ir H - dydžiai. Sąrašas yra pabruktas linijomis, kurių padėtis atitinka kodo lauko persijungimą, ir susijusiomis su iliustracijomis parodytomis FIG. 4C. Stulpelyje ch dešinėje pateikta dešimtainė eksponentės dydžio išraiška, o toliau - H dydžio eksponentės eh dešimtainės išraiškos. Kituose dviejuose stulpeliuose parodyti dešimtainis ir dvejetainis mantisės dydžio m atspindėjimai, atitinkamai. Toliau seka H - dydžio mantisės mh dešimtainis pavaizdavimas. Reikia pabrėžti, kad mantisė kinta nuo -1 iki -2 paveiksluose FIG. 5A, B, kur sh yra „0“, ir nuo +1 iki +2 paveiksluose FIG. 5C, D, kur dydis sh yra „1“. Kraštiniame dešiniajame stulpelyje parodytas dešimtainis H - dydis gautas iš binarinio rinkinio kairėje atlikus transformacijas.

Kaip matyti iš viršutinės FIG. 5B dalies, atspindėjimas „1“, kai po reikšmingiausio bito seka visi „0“ atitinka slankaus kabelio „0“.

Kaip seka iš binarinių skaičių sekos FIG. 4C, esančios centrinės linijos LIN kairėje, iš septynių reikšmingų bitų mažiausiai keturi bitai yra padalinti eksponentei ir mantisei, likusieji trys reikšmingiausi rezervuoti ženklo bitams ir kodo bitui. Iš stulpelių e ir m matome, kad viršutinėje dalyje kur eksponentės ženklas se ženklas yra neigiamas, t.y. „0“, ir kodo reikšmė c yra „1“, dydis e užima tris bitus ir dydis m vieną bitą (pirmieji du „0“ gali būti praleisti). Tai parodyta ir FIG. 4C, e sekcijoje. Sekančioje dalyje, kurioje eksponentės ženklas dar neigiamas, bet kodo reikšmė c yra „0“, dydis e užima tik vieną bitą, o mantisė - tris bitus. Tai atspindėta FIG. 4, pateikiant e atspindėjimui keletą žingsnių, duotų šioje dalyje. Dar sekančioje dalyje, dydis e užima tik vieną bitą, o mantisė - tris bitus ir t.t.

Kaip parodyta atspindėjimo dalyje, eksponentė e turi daug žingsnių sekcijoje, kai c=0 ir se=0, ir tik kelis žingsnius sekcijoje, kai c=1 ir se=1. Sekcijoje e+bias, esančioje sekcijos e kairėje, ištinėmis linijomis parodyti dydžiai e kiekviename ketvirtyje. Trūkia linija parodyta, kaip dydžiai e habs gaunami iš e+bias, invertuojant kiekvieną antrą sekcijos ketvirtį (eh atspindi e dydį su eksponentės ženklu).

Tokiu būdu, kodo bitas arba bitai priklauso nuo prieš einančio eksponentės ženklo bitų ir, gali būti sakoma, kad todėl kad

į kodą yra įjungti ir ženklų bitai galima turėti labai ribotą ilgį, t.y. kodas būtų trumpesnis, jeigu eksponentės ilgis priklausytų tiksliai nuo to kas paduodama į kodo lauką.

Mantisės ilgis priklauso nuo eksponentės taip, kad kai eksponentė ilga mantisė yra trumpa ir atvirkščiai. Todėl mantisė neparodyta FIG. 4, tačiau matoma iš stulpelių m, m sąrašas FIG. 5A ir 5B, kur pirmasis m parodo dešimtainį dydžio žymėjimą, antrasis - skaitmeninį jo atitikmenį. Jeigu $s=1$, tada $mh=1,0+m$, kitur $mh=-2+m$, kaip buvo aprašyta aukščiau.

Aprašytame pavyzdyje, tam, kad išlaikyti iliustratyvumą buvo nagrinėjamas septynių bitų žodis. Žodžio ilgis gali būti pasirinktas ilgesnis, pavyzdžiui 32 bitų ilgio. Tokiu atveju optimalu turėti kečiamo ilgio kodą. Taip pat tikslinga, tam kad supaprastinti schemotechninę realizaciją, padaryti perėjimą nuo vieno padalinimo tarp mantisės ir eksponentės prie kito žingsniais, apimančiais bitų grupes, pavyzdžiui po keturis. Tokio priėjimo pranašumas būtų tai, kad reikėtų mažiau kodų perėjimui nuo vieno padalinimo prie kito. Tokio tikslumo dekoderio pavyzdys bus aprašytas žemiau.

FIG. 6

Detalesnė grandinės parodytos FIG. 2 realizacija parodyta FIG. 6. Įėj/išėjimo magistralės v_0, v_1, v_2 ir v_3 yra sujungtos su įėjimo/išėjimo buferiu 20, perduodančiu informaciją į vidinių magistralių v_{0i}, v_{1i}, v_{2i} ir v_{3i} įėjimus, kiekvieno išorinio signalo CLOCK, paduodamo į buferio taktavimo įėjimą invertuoto taktavimo signalo metu, ir vidinių išėjimo magistralių $v_{0out}, v_{1out}, v_{2out}$ ir v_{3out} išėjimą į įėj/išėjimo magistralės v_0, v_1, v_2 ir v_3 kiekvieno taktinio signalo metu, atitinkamai.

FIG. 7A, 7B, 7C

Grandinės 20 realizacija ir jos taktavimas parodyti paveiksluose FIG. 7A, 7B ir 7C. Įėjimo/išėjimo buferis 20 sudarytas iš įėjimo buferių kiekvienai įėjimo magistralei v_0, v_1, v_2 ir v_3 ir tokio paties skaičiaus išėjimo buferių. Tam, kad nebūtų valdymo problemų, buferių taktiniai signalai turi nepersikloti. Tai formuojama perstumiant signalo priekinius frontus.

Pirmosios taktinio signalo CLOCK, parodyto FIG. 7A, fazės (0) metu invertuotas, kaip pažymėta kairėje skliausteliais, FIG. 7B, iš išorinio generatoriaus per ALĮ vidinės grandinės į buferių įėjimus paduodamas signalas valdo jų perdavimo būsenas. Duomenys sklinda vidinėmis ALĮ grandinėmis kol neperduodami transformuotoje formoje į išėjimo magistralės. Šiuo laikotarpiu išėjimo magistralės yra blokuojamos.

Kaip parodyta FIG. 7B, taktinio signalo CLOCK, pažymėto skliausteliais, fazės 1 metu įėjimo buferiai grandinėje 20 nustatyti į blokavimo būseną ir ALĮ vidinės grandinės veikia kaip multiplekseris, valdantis signalus, paduodamus į išėjimo/išėjimo magistralės v_0, v_1, v_2, v_3 . Išėjimo buferiai grandinėje 20 yra nustatyti į perdavimo būseną.

Įėjimo/išėjimo buferių būsenos vykstant daugiacykliui komandai fazės 0 metu parodytos FIG. 7C. Įėjimo/išėjimo magistralių v_0, v_1, v_2, v_3 turinys, paduotas iš ALĮ, prieš tai buvusios fazės 1 metu, fazės 0 metu išsaugotas jose dėka talpų gali būti perduotas per įėjimo buferius, nustatytus į perdavimo būseną, į įėjimo magistralės $v_{0i}, v_{1i}, v_{2i}, v_{3i}$, o iš čia per vidines grandines į ALĮ, kol nebus transformuotas į sekančią būseną.

Pagal pateiktą realizaciją, tiksliai įėjimo sąrašas, gaunamas perrašant turinį į įėjimo sąrašą ir paduodant rezultatą į išėjimą, yra naudojamas skaičiavimams atlikti ir valdyti. Sąrašą gali sudaryti atsimintos panaudojimo funkcijos arba kito tipo dydžiai. Pirmasis elementas sąrašas yra komandos kodas, likusieji elementai yra komandos argumentai. Tokio varianto pasirinkimą apsprendė pateiktos aprašyme aparatūros funkcijų priklausymas nuo įėjimo duomenų. Vienok, signalai, įvertinantys paduodamą į magistralės v_{1i}, v_{2i}, v_{3i} informaciją, yra išduodami valdymo įrenginiui VĮ (žiūrėti FIG. 1A) į EXT įėjimą.

Valdymo įtaisas VĮ gauna informaciją, atspindinčią komandas ir/arba skaičius, iš bet kurios magistralės $v_{0i}, v_{1i}, v_{2i}, v_{3i}$ ir, jeigu skaičiai yra sveiki arba slankaus kablelio dydžiai, perduoda šią informaciją į ALĮ valdymo grandinę magistrale EXT.

Vienok, gali būti naudingi praktiniai pritaikymai, kuriuose funkcinis ALĮ veikimas nereikalingas, ir tada išorinė informacija, paduodama iš VĮ į ALĮ valdymo grandinę magistrale EXT, gali būti panaudota skaičiavimų valdymui.

Komandos yra atliekamos perrašant komandas vienu žingsniu arba keliais žingsniais iki nebus pasiektas galutinis rezultatas.

Kai vieno žingsnio perrašymas yra atliktas, toks pat komandos kodas gali būti paliktas kitiems žingsniams arba sąrašas gali būti perrašytas su nauju komandos kodu, einančiu prieš dydžius.

FIG. 6

Informacijos apdorojimo įtaisai, pateikti FIG. 6, atlieka skirtingas funkcijas:

Pirmasis komparatorius 21, sujungtas su magistralėmis v_{0i} ir v_{1i} palygina visą informaciją šiose linijose, t.y. $v_{0i} > v_{1i}$, $v_{0i} = v_{1i}$ ir išduoda palyginimo rezultatus, t.y. dviejų bitų dydį, į išėjimą cmp01, sujungtą su ALĮ valdymo įtaisu 27. Antrasis komparatorius 22, sujungtas su magistralėmis v_{1i} ir v_{2i} palygina visą informaciją šiose linijose, t.y. $v_{1i} > v_{2i}$, $v_{1i} = v_{2i}$ ir išduoda palyginimo rezultatus, t.y. dviejų bitų dydį, į išėjimą cmp02, sujungtą su ALĮ valdymo įtaisu 27. Trečiasis komparatorius 221, sujungtas su magistralėmis v_{2i} ir v_{3i} palygina visą informaciją šiose linijose, t.y. $v_{2i} > v_{3i}$, $v_{2i} = v_{3i}$ ir išduoda palyginimo rezultatus, t.y. dviejų bitų dydį, į išėjimą cmp23, sujungtą su ALĮ valdymo įtaisu 27. Ketvirtasis eksponentinis komparatorius 222 yra prijungtas prie magistralių v_{1i} ir v_{2i} , kiekviena iš jų yra 32 bitų magistralė, ir reikšmingiausias žodžių šiose magistralėse dalis, t.y. dalis se, c, e, t.y. visą informaciją išskyrus mantisę ir ženklą. Komparatorius gauna pranešimą apie se, c dalių kodavimą, kaip aprašyta aukščiau. Taip pat signalas B1sb1, iš tikslumo dekoderio PD, paduodamas į atskirą komparatoriaus 222 įėjimą.

Penktasis komparatorius palygina signalus B 1sb1 ir B 1sb2 ateinančius iš tikslumo dekoderių. Išėjimo signalas cmp-pre indikuoja kai signalai B 1sb1 ir B 1sb2 yra lygūs.

ALĮ valdymo grandinė 27 palyginimo rezultatus, gautus iš komparatorių išėjimų, paduoda į išėjimą A STATE, sujungtą su valdymo įrenginiu VĮ (žiūrėti FIG. 1A). Rezultatai, taip pat, gali būti naudojami įtaisų, kurie bus aprašyti žemiau, valdymui, tokiu, kad išrinkti žodžiai būtų perduodami į išėjimo magistralės v1out, v2out ir v3out jeigu tai reikalinga.

Nereikšmingumo dekoderis 23 yra prijungtas prie linijos v1i ir tikrina ar bitai magistralėje v1i yra „0“ arba „1“, ar žodis magistralėje v1i atspindi sveiką ar slankaus kablelio skaičių, atskirai tikrina ar bitai eksponentinėje ir mantisės dalyse yra „0“ ar „1“. Tai parodanti informacija yra išvedama iš B 1sb1, indikuojančio mažiausiai reikšmingų bitų grupę magistralės v1i įėjime. Šis signalas duoda informaciją ar žodis magistralėje atspindi sveikus ar slankaus kablelio dydžius, ir slankaus kablelio dydžiams parodo padalinimą tarp mantisės ir eksponentės dalių. Tikrinimų rezultatas perduodamas į išėjimą insig1, sujungtą su ALĮ valdymo grandine 27.

Nereikšmingumo dekoderis 24 yra prijungtas prie linijos v2i ir tikrina ar bitai magistralėje v2i yra „0“ arba „1“, ar žodis magistralėje v2i atspindi sveiką ar slankaus kablelio skaičių, atskirai tikrina ar bitai eksponentinėje ir mantisės dalyse yra „0“ ar „1“. Tai parodanti informacija gaunama iš B 1sb2, indikuojančio mažiausiai reikšmingų bitų grupę magistralės v2i įėjime, ir duodančio tokio pat tipo informaciją kaip signalas B 1sb1. Šis testų rezultatas perduodamas į išėjimą insig2, sujungtą su ALĮ valdymo grandine 27.

Nereikšmingumo dekoderis 25 yra prijungtas prie linijos v3i ir tikrina ar bitai magistralėje v3i yra „0“ arba „1“, ar žodis magistralėje v3i atspindi sveiką ar slankaus kablelio skaičių, atskirai tikrina ar bitai eksponentinėje ir mantisės dalyse yra „0“ ar „1“. Tai parodanti informacija gaunama iš B 1sb1. Šis testų rezultatas perduodamas į išėjimą insig3, sujungtą su ALĮ valdymo grandine 27.

Kiekvienas nereikšmingumo dekoderis, taip pat, valdomas signalu c53, kuris bus aprašytas žemiau, nagrinėjant ALĮ valdymo grandinę 27.

Nereikšmingumo dekoderių įvedimo tikslas - tai informacijos viename iš įėjimų, naudojamos būsimuose skaičiavimuose, reikšmingumo nustatymas. Ši informacija paduodama į valdymo įrenginio VĮ magistralę A STATE. Paskutinis dalykas, kad mažiausiai vienas, vieno iš nereikšmingumo dekoderių, įėjimas, pavyzdžiui indikuojantis visus „0“, gali būti ALĮ valdymo grandinės 27 panaudotas kitoms operacijoms atlikti tokioms kaip, iteracijų skaičiavimas.

Komandų dekoderis 26 yra prijungtas prie v0i. Magistrale v0i perduodamos komandos, ALĮ vykdant aritmetines operacijas. Magistralės v0i dekodavimo rezultatas yra perduodamas į ALĮ valdymo grandinę 27 per išėjimus ins, komandų arba duomenų perdavimo metu. Įtaisai 23 - 26 valdomi signalais B 1sb1, B 1sb2 ir c5, suformuotais grandinėse PD1, PD2 ir 27 atitinkamai.

SUDĖTIES IR ATIMTIES ĮTAISAI

Pirmasis sudėties įtaisas 28, prijungtas prie magistralių v1i, v2i, atlieka sudėtį v1i+v2i ir perduoda rezultatą į atskirą išėjimą a1. Galimas rezultato perkėlimo signalas turėtų būti perduodamas į išėjimą gr a1, sujungtą su ALĮ valdymo grandine 27. Atimties įrenginys 29, prijungtas prie v1i ir v2i, atlieka atimtį v1i-v2i ir perduoda rezultatą į atskirą išėjimą a2. Galimas rezultato perkėlimo signalas turėtų būti perduodamas į išėjimą gr a2, sujungtą su ALĮ valdymo grandine 27. Antrasis sudėties įtaisas 30, prijungtas prie magistralių v1i, v2i, atlieka sudėtį v1i+2*v2i ir perduoda rezultatą į atskirą išėjimą a3. Galimas rezultato perkėlimo signalas turėtų būti perduodamas į išėjimą gr a3, sujungtą su ALĮ valdymo grandine 27.

Kiekvienas sudėties įtaisas yra valdomas signalu c5, ateinančiu iš ALĮ valdymo grandinės 27, kuri bus aprašyta detaliau, ir signalu B 1sb1. Šis signalas duoda informaciją, atspindinčią įėjimo magistralėse esančių žodžių tipą, t.y. ar žodis atspindi sveiką ar slankaus kablelio dydį, ir išlygina slankaus kablelio dydžių, esančių magistralėse v1i ir v2i, mantisės ir eksponentės dalis. Ši informacija yra esminė, nes diskretūs dydžiai, t.y. sveiki dydžiai, išreiškiami kaip viena visuma, o tolydūs dydžiai, t.y. slankaus kablelio dydžiai, kaip atskiros mantisės ir eksponentės dalys.

Pirmoji ženklo ir eksponentės ženklo grandinė 35 išrenka ženklą, eksponentės ženklą, reikšmingiausią mantisės bitą ir du mažiausiai reikšmingus bitus magistralės v1i žodyje ir perduoda informaciją per išėjimą sign1 į ALĮ valdymo grandinės 27 įėjimą. Antroji ženklo ir eksponentės ženklo grandinė 36 išrenka ženklą, eksponentės ženklą, reikšmingiausią mantisės bitą ir du mažiausiai reikšmingus bitus magistralės v1i žodyje ir perduoda informaciją per išėjimą sign2 į ALĮ valdymo grandinės 27 įėjimą. Taip pat, trečioji ženklo ir eksponentės ženklo grandinė 37 gali būti prijungta prie magistralės v3i ir išėjimu sign3 sujungta su ALĮ valdymo grandinės 27 įėjimu. Turi būti pažymėta, kad grandinės 35 ir 36 yra būtinos, kai ženklas, eksponentės ženklas ir mantisės bitai perduodami kaip žodžio, perduodamo magistralėmis v1i ir v2i, dalis. Įtaisai 35, 36 naudojami ALĮ valdymo grandinės esant slankaus kablelio žodžiams magistralėse. Grandinė 37 gali būti praleista eilėje ALĮ pritaikymų. Ženklo ir eksponentės ženklo grandinės eilėje ALĮ pritaikymų gali būti prijungtos (neparodyta) prie magistralės v0i.

ALĮ valdymo įrenginys 27, apdorojęs įėjimo signalus, paduoda skaitmeninius išėjimo signalus į išėjimo magistralės c1, c2, c3, c4, c5. ALĮ valdymo įrenginį geriausia realizuoti kaip loginių elementų matricą. Valdymo signalų pavyzdys bus aprašytas žemiau.

OPERANDU ĮTAISAS

Dviejuose operandų įtaisuose r1 ir r2, kiekvienam magistralių v1i ir v2i operandui, yra atliekami visi aritmetiniai veiksmai. Kiekvienas operando įtaisas turi keletą informacinių įėjimų, trys iš jų a1, a2 ir a3 sujungti su atitinkamu įtaisu 28, 29, 30, du - su magistralėmis v1i ir v2i, ir trys su atitinkamų tikslumo dekoderių PD1 arba PD2 išėjimu. Kiekvienas informacinis įėjimas yra magistralė, susidedanti iš to paties skaičiaus linijų kaip ir kiekviena įėjimo magistralė v0i, v1i, v2i, v3i.

Operando įtaiso r1 išėjimas yra išėjimo magistralė v1out,

operando įtaiso r2 išėjimas yra išėjimo magistralė v2out.

Pirmasis ALĮ valdymo įtaiso 27 išėjimo signalas c1 yra sudėtinis vidinių elementų operando įtaise r1 valdymo signalas. Kiti įtaiso r1 valdymo signalai paduodami iš tikslumo dekoderio PD1. Antrasis ALĮ valdymo įtaiso 27 išėjimo signalas c2 yra sudėtinis vidinių elementų operando įtaise r2 valdymo signalas. Kiti įtaiso r2 valdymo signalai paduodami iš tikslumo dekoderio PD2. Trečiasis išėjimo signalas c3 yra sudėtinis signalas, perduodamas į trečiąjį operando įtaisą r3, ketvirtasis išėjimo signalas c4 valdo polinomo, kurio formavimas bus aprašytas žemiau, grandinę 31. Penktasis išėjimas naudojamas valdyti tikslumo dekoderius PD1 ir PD2, nereikšmingumo grandines 23 - 25, sudėties įtaisyse 28, 30, atimties įtaisą 29, ir ženklų ir eksponentės ženklų grandines 35 - 37.

Operando įtaisyse r1 ir r2 yra vienodos konfigūracijos ir, todėl detaliau bus aprašytas vienas iš jų r2. Operando įtaisas yra suskirstytas į dvi dalis - į eksponentinę dalį OPe ir mantinės dalį OPm. Mantinės dalies OPm išėjimas n1, eksponentės dalies OPe išėjimas n2. Išėjimai n1 ir n2 prijungti prie išėjimo magistralės selektoriaus, taip pat vadinamo bitų grupės selektoriumi, M1 įėjimo. Generatorius 40, formuojantis tikslumo grupę iš keturių „0“ arba keturių „1“ pasirinktinai, ir valdomas signalu c2 11, ateinančiu į generatorių, ir, taip pat, prijungtu prie papildomo selektoriaus M1 įėjimo.

Keturi operando įtaiso įėjimai yra susiję su eksponentės dalimi OPe, t.y. v2i ir B bias, B coincr2, B codecr2, ateinantys iš tikslumo dekoderio PD2, ir taip pat įėjimą n8 iš mantinės dalies. Įėjimas v2i sujungtas su inkrementoriaus INC įėjimu, o dekrementoriaus DEC įėjimas su vienu iš penkių žodžio selektoriaus M2 įėjimų. INC ir DEC išėjimai n5 ir n6 yra prijungti prie dviejų žodžio selektoriaus M2 įėjimų. B coincr2 ir B codecr2 prijungti prie kitų žodžio selektoriaus M2 įėjimų.

Operando įtaiso r2 komponentė INC, taip pat turi perkėlimo išėjimus gr i2 ir C ei2, o komponentė DEC - perkėlimo išėjimus gr d2 ir C ed2, sujungtus su ALĮ valdymo grandine 27. (Operando įtaise r1, taip pat, yra komponentių INC ir DEC perkėlimo signalai C ei1, gr i1 ir C ed1, gr d1, paduodami į ALĮ valdymo grandinę 27.) Signalai perkėlimo išėjimuose generuojami pirmoje taktavimo ciklo pusėje. Selektoriai valdomi vėliau, pirmosios taktavimo ciklo dalies metu. Selektoriaus valdymo signalas formuojamas, įvertinant tai buvo ar ne perkėlimai inkrementoriaus arba dekrementoriaus grandinėse.

Didžiausias signalo, sklindančio į pirmąjį selektorių, vėlinimas yra didesnis nei pusė taktinio signalo periodo. Taigi, tokia realizacija papildomo vėlinimo neįneša. Vienok, tipinis vėlinimas, bendru atveju, yra mažesnis už pusę ciklo. Selektoriai nesukelia jokių signalo laikinių parametrų pakitimų.

Operando įtaiso žodžio selektoriaus M2 išėjimas n4 yra prijungtas prie žodžio selektoriaus M3, turinčio keturis įėjimus, pirmojo ir per inventorių INV prie antrojo įėjimų. Išėjimas n2 yra eksponentės dalies OPe išėjimas. Įėjimas B bias2 yra prijungtas prie žodžio selektoriaus M3 trečiojo įėjimo.

Mantinės dalis OPm turi penkis įėjimus, t.y. v2i, v1i, a1, a2 ir a3. Visi įėjimai yra sujungti su kiekvienu žodžio selektoriaus M4, turinčio penkis įėjimus ir išėjimą n8, įėjimu. Išėjimas n8 prijungtas prie žodžio selektoriaus M3 ketvirtąjį, susijusio su eksponentės dalimi OPe, įėjimo. Išėjimas n8, taip pat, yra sujungtas su vienu iš trijų žodžio selektoriaus M5 įėjimų, per neigiamo postūmio grandinę SH1, stumiančią binarinę informaciją, paduotą į įėjimą, per keturis žingsnius žemesniojo reikšmingumo kryptimi, t.y. padalina binarinę informaciją iš 16, ir prie trečiojo įėjimo per teigiamo postūmio grandinę SH2, stumiančią informaciją, paduotą į įėjimą, per keturis žingsnius aukštesniojo reikšmingumo kryptimi, t.y. padaugina binarinę informaciją iš 16.

Žodžio selektoriaus M5 išėjimas n7 yra tiesiogiai sujungtas su pirmuoju iš šešių žodžio selektoriaus M6 įėjimų; per neigiamo postūmio grandinę SH3, stumiančią binarinę informaciją, paduotą į įėjimą, vienu žingsniu žemesniojo reikšmingumo kryptimi, su antruoju įėjimu; per antrąją neigiamo postūmio grandinę SH4, stumiančią binarinę informaciją, paduotą į įėjimą, vienu žingsniu žemesniojo reikšmingumo kryptimi, su trečiuoju įėjimu, per teigiamo postūmio grandinę SH5, stumiančią binarinę informaciją, paduotą į įėjimą, vienu žingsniu aukštesniojo reikšmingumo kryptimi, su ketvirtuoju; per antrąją teigiamo postūmio grandinę SH6, stumiančią binarinę informaciją, paduotą į įėjimą, vienu žingsniu aukštesniojo reikšmingumo kryptimi, su penktuoju. Vidinis pastovaus žodžio generatorius 41, formuojantis žodį C word2 į valdymo įėjimą C2 10, yra sujungtas su atskiru žodžio selektoriaus M6 įėjimu. Pastovaus žodžio generatorius 41, valdomas signalu c2 10 gali duoti grupę, iš anksto apibrėžtų, vienetų ir nulių kombinacijų, pavyzdžiui visus „0“ arba „1“ arba žodį, parodantį dalinę informaciją. Generatorius gali atsimiti tiksliai ribotą kombinacijų, kurios išrenkamos signalu c2 10, žodžių skaičių. Žodžio selektoriaus M6 išėjimas yra mantinės dalies OPm išėjimas n1.

Mantinės dalis OPm atlieka šias funkcijas. Žodžio selektorius M4, įvertinęs kiekvieną v2i, v1i, a1, a2 arba a3, formuoja bazinį mantinės dydį n8 išėjime. Jis gali būti panaudojamas toks koks yra arba pastumiamas 1, 2, 3, 4, 5, 6 bitais į kairę (teigiamas postūmis) arba pastumiamas 1, 2, 3, 4, 5, 6 bitais į dešinę (neigiamas postūmis) arba atimamas iš žodžio C word2, formuojamo pastovaus žodžio generatoriumi 41, turinio. Eksponentės dalis OPe atlieka šias funkcijas. Jos išėjimu gali būti vienas iš šių signalų: nepastumtos mantinės dydis išėjime n8, v 2i+1, invertuotas v 2i+1, v 2i-1, invertuotas v 2i-1, B co-incr2,

invertuotas B co-incr2, B co-decr2, invertuotas B co-decr2, arba B bias2.

Operando įtaise r2, esantys žodžio selektoriai M2 - M6 atskirai valdomi, paduodamo iš ALĮ valdymo įtaiso 27 išėjimo c2, žodžiu c2. Tokiu būdu, žodžio selektorius M2 valdomas žodžio c2 dalimi c2 3, žodžio selektorius M3 valdomas žodžio c2 dalimi c2 2, žodžio selektorius M4 valdomas žodžio c2 dalimi c2 6, žodžio selektorius M5 valdomas žodžio c2 dalimi c2 5, žodžio selektorius M6 valdomas žodžio c2 dalimi c2 4. Postūmio grandinės taip pat valdomos žodžiu c2, SH1 ir SH2, sujungtos su selektoriaus M4 išėjimu n8, žodžio dalimi c2 8, postūmio grandinės SH3 - SH6 signalo dalimi c2 7, inkrementorius INC ir dekrementorius DEC signalo dalimi c2 9, pastovaus žodžio generatorius 41, formuojantis išėjimo signalą C word2, signalo dalimi c2 10, ir 0/1 generatorius 40 signalo dalimi c2 11.

Bitų grupės atskiras selektorius M1 valdomas kombinuotu daugiabičiu signalu, susidedančiu iš valdančio žodžio dalies c2 1, išėjimo kodo signalo B cde2, ateinančio iš tikslumo detektoriaus PD2, mažiausiai reikšmingo bito signalo B 1sb adj2, ir eksponentės išėjimo signalo B exp adj2, kur du paskutiniai paminėti signalai gauti iš signalų B 1sb2 ir B exp2, ateinančių iš tikslumo dekoderio PD2, po apdorojimo atitinkamose grandinėse 42A ir 42B. Apdorojimo grandinės 42A, 42B ir 43A, 43B, dažniausiai, duoda tokius pačius signalus kokie buvo paduoti iš susijusio tikslumo dekoderio į įėjimą. Vienok, yra keletas atvejų kai, atėjus perkėlimui iš inkrementoriaus INC arba dekrementoriaus DEC, išėjimo signalas būtų pastumiamas, kaip tai bus aprašyta žemiau. Šis postūmis būtų kontroliuojamas signalų c2 arba c1, ateinančių iš ALĮ valdymo grandinės 27, dalimis c2 5 arba c1 5, atitinkamai.

Tikslumo dekoderis PD1 (arba PD2) nustato visus nereikšmingus išėjimus, pavyzdžiui, susijusius su apdorojimo grandinėmis 42A, 42B (arba 43A, 43B) ir užpildytus visais nuliais, kai žodis v2i atspindi sveiką dydį, ir išėjimus tokius, kaip B exp2, B 1sb2 ir B cde (arba B exp1, B 1sb1 ir B cde2), kai žodis v2i (arba v1i) atspindi slankaus kablelio dydį. Detalės bus aprašyta žemiau.

FIG. 8

Bitų grupės atskiras selektorius M1 yra valdomas taip, kad išskirtų informacijos dalis, paduodamos į išėjimus n1 ir n2 ir iš generatoriaus 40, atitinkamai. Žodis, parodytas FIG. 3, t.y. atspindintis slankaus kablelio dydį, pavaizduotas viršutinėje FIG. 8 dalyje, padalintame stačiakampyje, demonstruojama kaip gali būti suformuotos bitų grupės. Atvaizdavimai s, se, c, e, m, msb ir 1sb parodo ženklą, eksponentės ženklą, kodo, eksponentės, mantisės, reikšmingiausio bito, mažiausiai reikšmingo bito dalis, atitinkamai. Selektoriaus M1 įėjimo informacijos padalinimo kombinacijų (mozaikų) pavyzdžiai pavaizduoti žodžio apačioje. Valdymo būdas, kaip įprasta, aprašomas mnemonikomis, kurios yra parodytos FIG. 8 kairėje.

Kiekviena mnemonika atspindima bitų paternais, susidedančiais, pavyzdžiui iš 5 arba 6 bitų. Bitų paternas paduodamas iš valdymo magistralės c2 1 į selektorių M1 (arba iš valdymo magistralės c1 1 į selektorių M1). Apdorojimo grandinės 42A ir 42B atlieka signalų B 1sb2 ir B exp2, ateinančių iš tikslumo dekoderių, postūmius ir perduoda signalus B 1sb adj2 ir B exp adj2, atitinkamai, į selektoriaus M1 įėjimą. Selektorius, tokiu būdu, valdomas keleto signalų, paduodamų į jo įėjimą, tačiau reaguoja ir į tam tikras, kombinuoto valdymo signalo, einančio per loginių elementų matricą, bitų kombinacijas, kiekvienai kombinacijai paduodant dalinę sudėtinio išėjimo signalo, sudaryto iš įėjimų n1, n2 ir 1/0 grandinės 40 išėjimo, kombinaciją, kaip matyti iš FIG. 8.

OPERANDŲ ĮTAISŲ r1 IR r2 SKIRTUMAI

Kaip buvo minėta, operandų įtaisai r1, yra tokios pačios konfigūracijos kaip ir r2. Operandų įtaisai r1 ir r2 skiriasi: įėjimai v1 ir v2 sukeisti vietomis. Konstanta C word yra priklausoma nuo konstantos C word2. Grandinė r1 valdoma signalu c1, padalintu į atskiras žodžio dalis c1 2 ir c1 11 ir ateinančiu iš ALĮ valdymo grandinės (vietoj signalo c2), ir tikslumo dekoderio PD1 signalu v1i se c1i, kuris yra eksponentės ženklą ir v1i, kaip įėjimo, kodo dalis (vietoj tikslumo dekoderio PD2 signalo v2i se c1i, kuris yra eksponentės ženklas ir v1i, kaip įėjimo, kodo dalis).

Tikslumo dekoderis PD1 paduoda išėjimo signalus B co-decr1, B co-icr1, B bias1, B exp1, B 1sb1, B msb1 ir B cde1 į operandų įtaisą r1 ir signalus B co max1, parodantį kai kodas atitinka maksimalų eksponentės ilgį, B co min1, parodantį kai kodas atitinka minimalų eksponentės ilgį, į ALĮ valdymo grandinę 27. Tikslumo dekoderis PD2 paduoda išėjimo signalus B co-decr2, B co-icr2, B bias2, B exp2, B 1sb2, B msb2 ir B cde2 į operandų įtaisą r2 ir signalus B co max1, parodantį kai kodas atitinka maksimalų eksponentės ilgį, B co min1, parodantį kai kodas atitinka minimalų eksponentės ilgį, į ALĮ valdymo grandinę 27. Išėjimo signalai B 1sb, B exp iš tikslumo dekoderių PD1 ir PD2, atitinkamai, gali būti pastumti, panašiu būdu kaip ir mantisė, kad gauti naują tikslumą, atspindimą išėjimuose, arba, kaip buvo aprašyta aukščiau, normalaus postūmio, kuris atliekamas valdymo grandinių 42A ir 42B arba 43A ir 43B valdymo signalo atitinkama dalimi c2 5 arba c1 5, būdu.

Kiekvienas tikslumo dekoderis yra sujungtas su signalu c5 3, indikuojančiu kai žodis įėjimo magistralėje v1i ir v2i, atitinkamai, atspindi sveiką ar slankaus kablelio dydį. Signalas c5 3 yra ALĮ valdymo įtaiso, valdančio grandinės 23 - 30, išėjimo signalo c5 dalimi. Tikslumo dekoderio išėjimo signalai bus aprašyti žemiau.

DAUGYBA IR DALYBA OPERANDO ĮTAISAS r3

Trečiasis operandų įtaisas r3, daugiklio/santykio funkcijos įtaisas, skiriasi nuo įtaisų r1 ir r2 ir daugiausia naudojamas daugybos ir dalybos operacijoms atlikti. Įtaisas r3 atlieka postūmius ir nustatymus būtinus daugybos ir dalybos operacijų metu. Jis valdomas kontroliniu žodžiu, gaunamu iš ALĮ valdymo įtaiso 27 išėjimo c3.

Daugiklio/santykio įtaisas r3 turi keletą įėjimų; į pirmąjį paduodama informacija iš magistralės v2i, į antrąjį - informacija

iš magistralės v3i ir trečiąjį - iš pastovaus žodžio generatoriaus 44 išėjimo, paduodančio išrinktą grandinės 27 išėjimo signalo c3 signalo dalies c3 4 veikimo metu, bitų kombinaciją iš keletos atsimintų generatoriuje.

Įtaisas r3 naudojamas vieno argumento įvertinimui sudėtingos skaitmeninės operacijos metu. Jo išėjimas prijungtas prie magistralės v3out.

Iėjimas v2i (magistralė) prijungtas prie žodžio selektoriaus M11 pirmojo įėjimo. Iėjimas v3i (magistralė) prijungtas tiesiai prie žodžio selektoriaus M11 antrojo įėjimo, per teigiamo vieno bito žingsnio postūmio grandinę SH11, valdomą signalo dalimi c3 5 - prie trečiojo įėjimo, per pirmąją neigiamo vieno bito žingsnio postūmio grandinę SH12 - prie ketvirtojo įėjimo ir per antrąją neigiamo vieno bito žingsnio postūmio grandinę SH13, - prie penktojo įėjimo, kur paskutinės dvi paminėtos postūmio grandinės valdomos signalais c3 3, B msb1, N1sb1. Generatoriaus 44 išėjimas yra sujungtas su šeštu žodžio selektoriaus įėjimu. Žodžio selektoriaus M11 išėjimas, yra sujungtas su antrojo žodžio selektoriaus M12 įėjimu ir išrenkamas ALĮ valdymo įtaiso 27 valdymo žodžio c3 dalimi c3 2. M12 yra magistralinis sujungimas, perduodantis selektoriaus M11 išėjimą į išėjimo magistralę v3out, ir yra nustatomas į padėtį „išjungta“ valdančiojo žodžio c3 dalimi c3 1. M12 valdymas bus aprašytas žemiau.

Buferio grandinė 26' yra prijungta prie magistralės v0i. Grandinė 26' yra magistralinis sujungimas, perduodantis v0i į savo išėjimą, tačiau gali būti valdomas, perjungiant į padėtį „išjungta“, valdančiojo žodžiu, paduodamu iš ALĮ valdymo grandinės 27. Taigi, grandinės 26' išėjime v0out paprastai bus ta pati informacija kaip ir magistralėje v0i.

SKAITMENINIO ALĮ VIDINIS TESTAS

Pateikiamas taip vadinamas polinomo generatorius 31. Polinomo generatorius formuoja ALĮ vidinio patikrinimo testo kodus. Generatorius 31 naudoja visumą uždarumų, apimančią visą informaciją magistralėse v0i, v1i, v2i, v3i kaip argumentus ir perduoda dydžius tikslo registrams į magistralės v0out', v1out', v2out', v3out'.

Vidinio testo metu, ALĮ valdymo įtaisas 27 valdo buferį 26' signalu c4, selektorių M1 operando įtaise r1 signalu c1 1, selektorių M1 operando įtaise r2 signalu c2 1, selektorių M12 operando įtaise r3 signalu c3 1, išjungdamas išėjimus magistralėse (v0out, v1out, v2out, v3out, atitinkamai), ir įjungia polinomo generatoriaus išėjimus, kurie paprastai yra išjunti. Tokiu būdu, v0out', v1out', v2out', v3out' yra polinomo generatoriaus išėjimai, kurie įjungiami perduoti informaciją į įėjimo magistralės v0, v1, v2, v3, vietoj magistralių v0out, v1out, v2out, v3out išėjimų, iš įtaisų 26', r1, r2 ir r3.

TRIŲ BŪSENŲ ĮTAISAS

Vidinių įtaisų 26', r1, r2 ir r3 išėjimo magistralės ir polinomo grandinės 31 išėjimo magistralės v0out', v1out', v2out', v3out' yra prijungtos prie skirtųjų trijų būsenų įtaiso 39 įėjimų. Įtaiso 39 struktūra panaši į buferio, parodyto FIG. 7A-7C, struktūrą ir yra valdomas ALĮ valdymo grandinės 27 signalu c4, kuris vidinių įtaisų buferius įjungia į perdavimo režimą, o polinomo grandinės 31 vidinių išėjimo magistralių buferius išjungia, t.y. nustato į informacijos blokavimo režimą. Tai yra normalios ALĮ darbo sąlygos ir tik testo metu buferiai perjungiami kitokiu būdu.

FIG. 9 IR TIKSLUMO DEKODERIAI PD1 IR PD2

FIG. 9 parodytas tikslumo dekoderių, kai žodis atspindi slankaus kabelio dydį įėjimo magistralėje v1i arba v2i, išėjimo signalas. Tikslumo dekoderis darbas su slankaus kabelio dydžiais bus aprašytas su FIG. 9 parodytu pavyzdžiu.

Tikslumo dekoderio argumentai, kurie aprašinėjami žemiau, tai eksponentės ženklas ir kodo dalis, aprašanti slankaus kabelio dydį, ir valdymo signalai parodantys, kad žodis atspindi paduotą slankaus kabelio dydį. Jis yra 1+4 bitų ilgio.

Skirtingi kodai kodo bitų lauke atitinka skirtingiems tikslumams (mantisės ir eksponentės ilgiai). Eksponentės mažiausiai reikšminga grupė 1sb (žiūrėti FIG. 8) kinta kartu su tikslumo kitimu. Ši grupė, pavyzdį iliustruojančiame žodyje FIG. 9 sudaro bitų patarną 0100. Binariniame signale B 1sb skiriama po vieną bitą kiekvienai tikslumo grupei, t.y. aštuoni bitai, ir jie yra „0“ visoms grupėms, išskyrus mažiausio reikšmingumo grupę.

Išėjimas n8, taip pat, yra sujungtas su vienu iš trijų žodžio selektoriaus M5 įėjimų, per neigiamo postūmio grandinę SH1, stumiančią binarinę informaciją, paduotą į įėjimą, per keturis žingsnius žemesniojo reikšmingumo kryptimi, t.y. padalina binarinę informaciją iš 16, ir prie trečiojo įėjimo per teigiamo postūmio grandinę SH2, stumiančią informaciją, paduotą į įėjimą, per keturis žingsnius aukštesniojo reikšmingumo kryptimi, t.y. padaugina binarinę informaciją iš 16. Kodas ir jo invertuota forma atitinka tą patį tikslumą, kur atitinkami eksponentės ženklai yra papildomi, kaip parodyta lentelėje žemiau.

grupė 1sb	kodas įtrauk.	eksp. ženklą
0(įtrauk. žodžio 1sb)	nenaudojamas	nenaudojamas
1	1 1111	+0 0000
2	1 1110	+0 0001
3	1 1101	+0 0010
4	1 1100	+0 0011
5	1 10--	+0 01--
6	1 0---	+0 1
7(įtrauk. žodžio msb)	nenaudojamas	nenaudojamas

Kode, parodytame aukščiau, eksponentės ženklas pavaizduotas kaip vienas elementas. Kai šis elementas yra būsenoje „teisinga“ visi kiti bitai naudojami taip kaip yra, bet kai jis yra būsenoje „neteisinga“ jie visi invertuojami.

Kaip minėta aukščiau, tikslumo dekoderis perduoda loginių kintamųjų sekas, t.y. keletą išėjimo žodžių, kurie paduodami į skirtingus ALI įtaisus, tokius kaip operando įtasis, ALI valdymo grandinę ir t.t. Žodžiui, atspindinčiam slankaus kabelio dydį, signalas B 1sb gali perduoti bitą „teisinga“ tiksliai vienai iš 1 - 6 grupių, esant laisviems bitams būsenose „neteisinga“. Grupėms 0 ir 7 visada paduodamas bitas „neteisinga“.

Aštuonių loginių kintamųjų seka (viena - kiekvienai tikslumo grupei) indikuojama, priklausančiu tikslumo grupėms iš kodo/eksponentės dalies, bitu, esančiu būsenoje „teisinga“, kaip tai parodyta B exp FIG. 9.

Parodytame pavyzdyje, dėka keičiamo kodo ilgio, galimos trys reikšmingiausio eksponentės bito pozicijos. Kaip pavaizduota FIG.9, B msb, yra trys loginiai kintamieji, parodantys vieną iš pozicijų su bitu būsenoje „teisinga“ (kiti bitai būsenoje „neteisinga“. Taigi, B msb yra 001, parodydamas, kad kodo ilgis yra maksimalus, toks kaip grupėms 1sb ir nuo 1 iki 4 (žiūrėti lentelę aukščiau). 010 indikuotą grupę 5, o 100 - grupę 6.

Pavyzdyje, parodytame FIG. 9, B cde yra aštuonių kintamųjų, koresponduojančių su visais bitais dviejose reikšmingiausiose tikslumo grupėse, funkcija. B cde yra „teisinga“ („1“) nuo mažiausiai reikšmingo kodo bito iki eksponentės ženklo.

B bias yra 32 bitų ilgio žodis, naudojamas slankaus kabelio dydžio atspindėjimo reguliavimui. visi jo bitai, išskyrus mažiausiai reikšmingą eksponentės bitą, yra būsenoje „neteisinga“ („0“).

Tikslumo dekoderis, taip pat, formuoja didėjančius ir mažėjančius kodus B co incr ir B co decr, atitinkamai, kurie yra naudojami kintamo tikslumo operacijose. Tiksliai pirmieji bitai, praplėsti virš eksponentės ženklo ir žodžio kodo dalies, yra parodyti signalams B co incr ir B co decr. Vienok, optimaliausias žodžio ilgis yra 32 bitai. Likusiųjų bitų geriausia būseną yra „0“.

Du loginiai kintamieji B co max ir B co min parodo kokią, maksimalaus ar minimalaus ilgio eksponentę, atspindi kodas. Pavyzdyje, parodytame FIG. 9, mažiausiai reikšmingas bitas paduodamas į 4 grupę. Tada, abu loginiai kintamieji B co max ir B co min yra „0“. Jeigu mažiausiai reikšmingas eksponentės bitas buvo paduotas į 1 grupę, tada B co min buvo „1“, o jeigu buvo paduotas į 6 grupę, tada B co max buvo „1“.

Signalai, parodyti FIG. 9, gaunami slankaus kabelio dydžiui. Sveikam dydžiui visi signalai B 1sb, B exp, B msb ir B cde yra būsenoje „neteisinga“, t.y. visi lygūs „0“. Reikia pažymėti, kad galima panaudoti kitą bitų kombinacijų rinkinį, parodantį sveiką dydį. Rinkiniui keliama sąlyga, kad laisvai pasirinkta, dalinė šių signalų bitų kombinacija atspindėtų sveiką dydį.

Tikslumo dekodero realizacija bus aprašyta sąryšyje su FIG. 13.

SLANKAUS KABELIO DYDŽIŲ SUDĖTIS, PAVYZDYS

Sveikų skaičių daugybos pavyzdys buvo duotas sąryšyje su FIG. 2 aprašymu. Čia bus pateiktas dviejų slankaus kabelio dydžių sudėties pavyzdys, iliustruojantis ALI veikimą.

Kaip buvo minėta, tolydaus kodavimo schema nėra perteklinė, t.y. nepriedami jokie duomenys klaidai detektuoti. Taip pat, įprastos operacijos, tokios kaip išlyginimas ir tikslumo reguliavimas, nerealizuojamos tiesioginiu būdu. Trys būsenos magistralėje v0i yra paskirtos šiai problemai spręsti. Jos apsprendžia kaip operandai magistralėje v1i ir v2i, žemiau vadinami a - operandu ir b - operandu, yra koduojami. Šios būsenos yra:

N operandai a ir b normalizuoti.

A operandas a - keičiamo kodo, jis nenormalizuotas.

B operandas b - keičiamo kodo, jis nenormalizuotas.

MANTISĖS INTERPRETACIJA

Fig. 10 parodytas mantinės bitų serijos 1010 normalizuotam ir keičiamam kodams interpretacijos pavyzdys. Virtualus „1“ paduotas normalizuotam kodui duoda rezultatą 1,625, kai be virtualaus vieneto keičiamam kodui rezultatas būtų 0,625.

FIG. 11 IR 12. IŠLYGINIMAS

Pirmasis tolydžios sudėties, t.y. dviejų slankaus kabelio dydžių sudėties, žingsnis yra operandų įėjimo magistralėje išlyginimas, kai skiriasi jų eksponentinės dalys. Jeigu sudedami dydžiai a ir b yra $a=1,5 \cdot 2^{31}$ ir $b=-1,125 \cdot 2^{33}$ tai, dydžio a eksponentinė dalis yra mažesnė už dydžio b, ir dydžių eksponentės turi būti išlygintos, kaip pavaizduota FIG. 11 ir 12 dydžiui a pirmuoju ir antruoju išlyginimo ciklais, prieš sudėties operacijos atlikimą.

Pavyzdys parinktas taip, kad naudojami trys dviejų operandų išlyginimo variantai. Dydžio a eksponentė padidinama 1, o mantisė padalinama iš 2 (pastumiant per vieną žingsniu žemesnio reikšmingumo kryptimi). Tai atliekama du kartus, kad gauti tokią pat kaip dydžio b eksponentinę dalį.

Pavyzdys parinktas taip, kad apimtų tris dviejų argumentų išlyginimo variantus:

1. Pradinis dydis išlyginimo cikle yra atspindėtas binarine kombinacija, kurioje eksponentė didesnė už galimą c kodą ir eksponentės ženklas yra neigiamas. Pažymėsime, kad skaičiaus tikslumas priklauso nuo eksponentės ženklo. Eksponentės tikslumas apdorojimo metu turi būti padidintas. Eksponentinė dalis padidinama vienetu, tada mantisė pastumiama į kairę atitinkamu skaičiumi žingsnių.

2. Pradinis eksponentės dydis išlyginimo cikle didesnis už galimą c kodą ir eksponentės ženklas yra teigiamas. Eksponentės tikslumas apdorojimo metu turi būti sumažintas. Eksponentinė dalis padidinama vienetu, tada mantisė pastumama į dešinę atitinkamu skaičiumi žingsnių.

3. Pradinis eksponentės dydis išlyginimo cikle neviršija galimo c kodo. Eksponentė didinama, o mantisė dalinama iš dviejų.

Pirmame dydžio a išlyginimo cikle, t.y. keičiant bitų atspindėjimą, parodytą aukščiau į atspindėjimą parodytą žemiau FIG. 11, tikslumo kodas, ryšium su tuo, kad eksponentė yra didinama ir yra maksimali, t.y. visi bitai yra „1“, padidinamas vienu bitu iš dešinės ir perkėlimo signalas paduodamas į tikslumo kodą, kuris taip pat padidinamas. Po šių operacijų kodas interpretuojamas, kaip vienu bitu ilgesnis kodas (žiūrėti grupės 5 kodą lentelėje). Tai reiškia, kad eksponentės ilgis sumažinamas vienu bitu iš kairės pusės ir vienetu padidinama tikslumo grupė (4 bitai) iš dešinės pusės. Eksponentės dydis taip pat padidinamas vienetu. Mantisė pirmiausia yra pastumta per keturis žingsnius į dešinę (tikslumas sumažėjo) ir tada pastumta vienu žingsniu į dešinę (padalinta iš 2). Kai mantisės grupė stumiama vienu žingsniu žemesniojo reikšmingumo kryptimi, virtualus „1“ įstumiamas į mantisės dalį, kurioje, po pirmojo išlyginimo ciklo reikšmingiausioje tikslumo grupėje, bus dvejetainė kombinacija 1100.

APARATŪRINIS IŠLYGINIMO REALIZAVIMAS

Išlyginimo operacija atliekama vieno ALI ciklo metu operando įtaise r1, nuskaitant dydį iš magistralės v1i. Eksponentės dalis yra didesnis nei galimas dydžio a atspindėjimas. Eksponentės ženklas teigiamas, tai reiškia, kad visi eksponentės bitai yra lygūs „1“. Kai eksponentė padidinama inkrementoriaus INC, visi bitai tampa lygūs „0“ ir inkrementorius INC formuoja perkėlimą gr i1 į ALI valdymo grandinę 27, kuri pakeičia išėjimo signalo c1 5 binarinę dalį taip, kad juo valdomi apdorojimo įtaisai 43A ir 43B pastumtų signalą, atėjusį iš tikslumo dekoderio PD1, į kairę. Perkėlimo signalas gr i1, taip pat, pernešamas į kodo dalį.

Panašaus tipo aprašymas tinka ir dekrementoriui DEC tada, kai atliekamas kodo mažinimas, tačiau, kai vykdomas perkėlimas gr dj, kur j yra 1 arba 2, signalo dalis c j5 valdo apdorojimo įtaisus 43A ir 43B arba 42A ir 42B, atitinkamai, taip, kad būtų atliktas postūmis į kairę. Reikia pažymėti, kad inkrementoriai ir dekrementoriai formuoja išėjimus pirmojoje pirmosios taktinio signalo periodo pusės dalyje ir šie valdymo signalai gali būti paduoti į apdorojimo įrenginius ankščiau, šios periodo dalies metu. Visų operando įtaisų selektorių valdymas yra atliekamas antroje pirmosios taktinio signalo periodo pusės dalyje. Tikslumo detektorių išėjimai nesikeičia antroje taktinio signalo dalyje. Juose, kaip ir vidinėse įėjimo magistralėse v0i, v1i, v2i, v3i yra pastovūs dydžiai. Kaip minėta aukščiau, pirmojoje taktinio signalo dalyje įėjimo buferiai, magistralės sujungime 20, praleidžia duomenis iš išorinių magistralių v0, v1, v2, v3, atitinkamai, kol išėjimo buferiai izoliuoja duomenis nuo operando įtaisų ir išorinės magistralės. Antroje taktavimo ciklo dalyje įėjimo buferiai saugo dydžius vidinėse įėjimo magistralėse v0i, v1i, v2i, v3i, tol kol nauji išėjimo duomenys perduodami į išorinę magistralę. Tai reiškia, kad tikslumo dekoderiai įtakojami naujais išėjimo duomenimis sekančiame taktinio signalo periode.

Mantisė, postūmio į dešinę per vieną tikslumo grupę įtaisu SH2, esančiu operando įtaise r1, pastumama per vieną tikslumo grupę į dešinę (t.y. padalinama iš 16). Mažiausiai reikšminga eksponentės tikslumo grupė parodoma bito vektoriaus signalu B 1sb1, gaunamu iš tikslumo dekoderio PD1. Postūmis į dešinę šioje grupėje atliekamas apdorojimo įtaisais 43A ir 43B. Tam, kad pastumti mantisę vienu žingsniu toliau nuo naujos padėties su postūmio įtaisu SH5 (dalinančiu iš 2), vektorius B 1sb1 priderinamas pastumti į dešinę įtaisais 43A ir 43B. Priderintas vektoriaus signalas B 1sb adj1 naudojamas iš mantisės dalies OPm ir eksponentės dalies OPe išskirti išėjimą. Pridėtinė grupė iš keturių „0“, gaunama iš grandinės 40, yra įterpiama į r1 kaip nauja mažiausiai reikšminga eksponentinės dalies tikslumo grupė, tam kad ją užpildyti (žiūrėti FIG. 8 cef_m/cet_m).

Normalizuotamo kodo virtualus bitas yra lygus „1“ ir stumiamas į naują reikšmingiausią mantisės padėtį, atitinkančią normalizuotos teigiamos mantisės dalybą $((1,0+m)/2)$. Nauja mantisė nenormalizuota ir dydis a bus keičiamo kodo dydis (t.y., kurio virtualus bitas tarp eksponentės ir mantisės dalių lygus „0“).

Sekančio išlyginimo, parodyto FIG. 12, ciklo metu, keisti tikslumo nereikia, t.y. bitų skaičius eksponentinėje dalyje nekeičiamas. Eksponentė padidinama 1 ir kodas c išlieka nepakeistas. Perkėlimo signalas kodo keitimui šiame cikle neperduodamas, kadangi paduoto dydžio eksponentė nėra galimai didžiausia.

Mantisė yra pastumama per vieną žingsnį į dešinę (SH5). Kadangi postūmis atliekamas su keičiamu kodu, tarp eksponentės ir mantisės dalių nėra virtualaus „1“, o virtualus „0“ yra įstumiamas ir dvejetainė reikšmingiausios tikslumo grupės kombinacija būtų 0110. Taigi po išlyginimo a yra keičiamo kodo dydis ir atspindimas kaip $0,374 \cdot 2^{33}$.

Tam, kad gauti normalizuotą mantisę, normalizuojami argumentai, o ne sudėties rezultatas. Taigi, $a+b$ rezultatas yra $0,375-1,125 = -0,750 \cdot 2^{33}$, kuris nėra normalizuotas rezultatas. Argumento normalizavimas, einantis po sudėties, duoda normalizuotą sumą $-1,5 \cdot 2^{32} = (2 \cdot (-0,750 \cdot 2^{33}/2))$.

Slankaus kabelio dydžių daugyba atliekama panašiai kaip ir, aukščiau aprašyta sąryšyje su FIG. 2, sveikų dydžių daugyba. Vienok, tam kad sumažinti vėlinimą, naudojami du bitai vietoj vieno.

Daugybės ciklo metu lygybė mantisėms yra $sma \cdot smb = smv \cdot 2i \cdot smv \cdot 3i$, kur sm parodo, kad mantisės dydis sukomponuotas iš ženklo, virtualaus bito ir mantisės bitų.

Žodyje v1i nėra eksponentės ženklo, tikslumo kodo arba eksponentės dydžio ciklo metu. Todėl mantisė yra ilgesnė nei tikrasis operacijos tikslumas. Priekiniai mantisės bitai yra koduojami tokiu būdu, kad pilnas žodis v1i yra diskretus dydis

lygus daliai mantisei. Ženklo bite yra teisingas sandaugos dydis (ženklas).

Žodis v2i yra daugiklis. Jis nėra keičiamas.

Žodis v3i aprašomas eksponentės ženklu, tikslumu ir eksponentės dydžiu. Vienok, mantisė yra papildyta ženklo bitu, leidžiančiu gauti didesnę tikslumą. Mantisės dalis žodyje yra vienu bitu ilgesnė. Mantisė šiuo atveju yra užkoduota kaip diskretus dydis be ženklo bito. Ženklo bitas gali būti išvestas iš žodžio v1i ženklo.

Daugybės ciklo metu žodis v1i yra stumiamas į dešinę. Žodis v3i taip pat stumiamas per du bitus į dešinę kiekvieno ciklo apsisukimo metu (SH13). Kai visi bitai buvo išstumti ciklas baigiasi. Šiuo atveju, rezultatas yra normalizuotas. Manoma, kad specialistai, remdamiesi aukščiau duotų pavyzdžių aprašymais, nesunkiai gali suprasti kaip ALĮ atlieka kito tipo operacijas.

FIG. 13. TIKSLUMO DEKODERIS

Tikslumo dekodero struktūra parodyta FIG. 13. Reikia pažymėta, kad gali būti atlikti pakeitimai siekiant pagreitinti skaičiavimus. Tačiau jie yra konstruktorinio pobūdžio ir čia nebus aprašomos. Tikroji grandinė gali turėti kitą nei parodyta FIG. 13 konfigūraciją, tačiau tai kas atlieka principines funkcijas išlieka kaip parodyta paveiksle. Įėjimo signalo vji, kur j lygus 1 arba 2, eksponentės ženklo ir kodo dalis yra prijungta prie perkodavimo grandinės 50, kuri, šioje principinėje tikslumo dekodero realizacijoje, į savo išėjimus paduoda vieną iš kodo variantų, įjungiančio eksponentės ženklą, parodytą lentelėje, pavyzdžiui kairįjį, ir padavus bet kurį variantą į įėjimą. Pasirinktinai, abu variantai gali būti paduoti į perkodavimo grandinės 50 įėjimą, bet visada ta pačia tvarka, pavyzdžiui kairysis ir dešinysis variantai, kaip parodyta lentelėje aukščiau. Išėjimas yra sujungtas su bitų sekos komparatoriumi 51, kuris kontroliuoja bitų įėjimuose kombinacijas ir formuoja signalą B msb, kodo ilgį, žiūrėti FIG. 8, t.y. vieno, dviejų ar trijų bitų ilgio kodas pateiktame pavyzdyje. Grandinė 51 formuoja signalą B cde.

Šešios bitų sekos komparatorių grandinės 52 - 57 yra, taip pat, prijungtos prie perkodavimo grandinės 50 išėjimo, kiekviena pritaikyta formuoti „1“ signalą, vietoj „0“ normalioje būsenoje, išėjime, kai jų dalinis kodas ir paternas perkodavimo grandinės 50 išėjimuose sutampa.

Kaip matyti iš FIG. 8, tikslai viena iš grandinių tuo pačiu metu duoda „1“, kitų išėjimuose yra „0“, t.y. „netiesinga“. Pirmasis ir paskutinis signalo B 1sb bitai visada yra „0“ būsenoje. Tokiu būdu, dvi linijos 58 ir 59 yra tiesiai sujungtos su linija, turinčia bitą „0“.

Linija 58, grandinių 52 - 57 išėjimai ir linija 59 sudaro magistralę B 1sb, kurioje išskiriama po vieną bitą kiekvienai tikslumo grupei. Magistralėje B bias išskiriama po keturis bitus kiekvienai tikslumo grupei. Kiekvieną grupę sudaro trys bitai, esantys „0“ būsenoje ir signalo B 1sb bitas kiekvienai daliai grupei. Tokiu būdu, kiekvienos grupės trys linijos yra sujungtos su „0“, o ketvirtoji prijungta prie atinkamos B 1sb išėjimo linijos. Išėjimas B co-min yra prijungtas prie grandinės 52 išėjimo, o išėjimas B co max - prie grandinės 57 išėjimo.

Išėjimo B exp bitų paternas sudarytas iš „0“ žemiausio reikšmingumo pozicijose, iki padėties kur kur išėjime B 1sb yra „1“, o tolimesni bitai yra „1“. Žemiausio reikšmingumo bitas, kuris yra sujungtas su linija 59 visados bus „0“, tačiau nuo sekančios pozicijos yra galimi „1“. Tokiu būdu, ARBA loginio elemento 60 vienas įėjimas prijungtas prie grandinės 57 išėjimo, o kitas prie grandinės 56 išėjimo. Grandinės 57 išėjimas prijungtas prie signalo B exp antrosios žemiausio reikšmingumo linijos. Loginio elemento ARBA 60 išėjimas prijungtas prie išėjimo B exp antrosios žemiausio reikšmingumo linijos. Taigi, grandinės 57 išėjime atsiradęs „1“ nustatytų į „1“ ir loginio elemento 60 išėjimą. Elemento 60 išėjime būtų „0“ tada, kai grandinių 57 ir 56 išėjimuose būtų „0“. Elemento ARBA 60 išėjimas yra prijungtas prie pirmojo kito elemento ARBA 61, kurio antrasis įėjimas prijungtas prie grandinės 55 išėjimo, įėjimo. Elemento 61 išėjimas prijungtas prie trečiosios žemiausiojo reikšmingumo išėjimo B exp linijos ir prie elemento ARBA 62 pirmojo įėjimo, kurio antrasis įėjimas yra prijungtas prie grandinės 54 ir t.t. Toks kaskadinis ARBA elementų jungimas duoda „1“ lygius visose pozicijose, turinčiose lygų arba aukštesnį, kaip išėjimo B 1sb „1“, pozicijos reikšmingumą.

Žodžio B co decr ženklas, eksponentės ženklas ir kodo dalis pirmąja linija sujungta su įtaisu duodančiu „1“, antrąja su invertoriumi, sujungtu su šeštąja B 1sb linija, trečiąja - su ARBA elementu 162, kurio įėjimai sujungti su penktąja ir šeštąja B 1sb linijomis, ir ketvirtoji ir penktoji linijos tiesiai sujungtos su B 1sb antrąja ir trečiąja linijomis atitinkamai. Žodžio B co decr linijos ženklas, eksponentės ženklas ir kodo dalis prijungtos prie bito užpildymo grandinės 66, pridedančios „0“, prie įėjimą paduodamo žodžio, taip, kad gauti 32 bitų žodį išėjime.

Žodžio B co incr ženklas, eksponentės ženklas ir kodo dalis pirmąja ir antrąja linijomis sujungtas su įtaisu duodančiu „1“, trečiąja - su invertoriumi 163 sujungtu su septintąja B 1sb linija, ketvirtąja - su ARBA elementu 164, kurio įėjimai sujungti su trečiąja ir ketvirtąja B 1sb linijomis, ir penktoji linija tiesiai sujungta su ARBA elementu 165, kurio įėjimai sujungti su B 1sb trečiąja ir penktąja linijomis. Žodžio B co incr linijos ženklas, eksponentės ženklas ir kodo dalis prijungtos prie bito užpildymo grandinės 67, pridedančios „0“, prie įėjimą paduodamo žodžio, taip, kad gauti 32 bitų žodį išėjime.

FIG. 14A - 14C. SUDĖTIES ĮTAISAS

Sudėties įtaiso, tokio kaip viena iš grandinių 28 - 29, pavaizduotų FIG. 6, struktūra parodyta FIG. 14A - 14C. Sudėties įtaisas turi du argumentus a ir b. Argumentas a paduodamas į magistralę v1i, argumentas b - į magistralę v2i. Šie argumentai koduojami priklausomai ar tai sveiki ar slankaus kablelio dydžiai. Signalas c5 1, kuris yra „0“ sveikiems dydžiams ir „1“ slankaus kablelio dydžiams iš grandinės 27, FIG. 6, išėjimo c5 ir magistralių B 1sb1 ir B 1sb2, kurių visi bitai „0“ - sveikam dydžiui ir vienas bitas nustatytas į „1“ - slankaus kablelio dydžiui, indikuoja šį faktą. Kaip parodyta FIG. 14A, sudėties įtaisas

binarinis įrenginys, sudarytas iš aštuonių kaskadu sujungtų sekcijų G0, G1, . . . G7, t.y. viena sekcija kiekvienai tikslumo grupei.

Kaip matyti iš aprašymo aukščiau, prieš sudedant slankaus kabelio skaičius svarbiausia išlyginti argumentus a ir b. Tai reiškia, kad B 1sb1 ir B 1sb2 turi būti lygūs. B 1sb1 yra signalas sumavimo įrenginiui valdyti. Signalas B 1sb1 pastumiamas vienu žingsniu mantisės kryptimi, prieš paduodant jį į sumatorius, inkrementorius, dekrementorius ir t.t., kadangi gali įtakoti perkėlimo signalo perdavimą reikšmingiausioje mantisės grupėje. Mažiausiai reikšmingas signalo B 1sb1 bitas šiais atvejais nenaudojamas.

Argumentai a ir b, susidedantys iš 32 bitų kiekvienas, yra suskirstomi į aštuonias grupes, kiekvienoje po keturis bitus. Kiekviena keturių bitų grupė a ir b paduodama į skirtingas sekcijas Gi, kur i -sveikas skaičius nuo 0 iki 7. Kiekviena magistralės B 1sb linija prijungiama prie atitinkamų sekcijų Gi, išskyrus G0. Linija c in paduota į sekciją G7 parodo kokią sudėties ar atimties operaciją turės atlikti sudėties įtaisas. c in yra „0“ -sudėčiai ir „1“ - atimčiai. Kiekviena sekcija formuoja pemešimo signalą c out, kuris paduotas į sekančią sekciją ir yra sujungtas su perkėlimo signalu c in i, ateinančiu iš prieš esančios sekcijos. Detalus aprašymas bus pateiktas žemiau.

Argumentų a ir b sudėtis atliekama atskirai, kiekvienoje sekcijoje G0 - G7, paduodant pemešimo signalus į sekančią sekciją seka iš G7 į G0, jeigu reikalinga. Slankaus kabelio dydžiui, jeigu reikalinga, mantisės perkėlimui gali būti panaudota magistralė c m. Ši magistralė kartu su linijomis c s, kurioje perduodamas ženklų bitas, ir c se, kurioje perduodamas eksponentės bitas, formuoja signalą gr a1 (arba gr a2 arba gr a3) paduodamą į grandinę 27 (FIG. 6). Grandinė 27 perduoda šią informaciją į valdymo įrenginį V1 magistrale A_{STATE}.

Signalas c5 1 parodo kokie yra argumentai a ir b, sveiki ar slankaus kabelio dydžiai, signalas c5 2 parodo koks yra kodas, normalus ar keičiamas argumentui a, signalas c5 3 parodo koks yra kodas, normalus ar keičiamas argumentui b. Šie signalai paduodami į sekciją G0, kuri atitinka reikšmingiausią tikslumo grupę, įjungiačią reikšmingiausio bito išpjovą (ženklą išpjovą). Tai dalinai priklauso nuo operandų a ir b formatų (t.y. ar jie abu yra sveiki ar slankaus kabelio dydžiai).

Trijų būsenų elementas 70 įjungia magistralę c m į nustatytą dydį, pavyzdžiui į visus „0“, kai operandai yra sveiki dydžiai, ir mantisės perkėlimas magistralėje neaktyvuojamas. Trijų būsenų įtaisas 70 valdomas signalu c5 3.

Sekcijos G1 struktūra parodyta FIG. 14B. Ji yra „supjaustyta“ į keturias sumavimo įtaiso išpjovas AS1 - AS4. Sekcijoje yra perkėlimo generatorius, formuojantis perkėlimo signalą cout į sekančią sekciją. Kiekvienai bitų grupei a i ir b i, sekcijoje Gi, yra atskiri bitai a ik ir b ik, atitinkamai sujungti su kiekviena atskira sumatoriaus išpjova ASk, kur k sveikas skaičius kintantis nuo 1 iki 4. Kiekviena sumatoriaus išpjova paduoda vieną sumos sum ik bitą į išėjimą sum output.

Kaip bus aprašyta detaliau FIG. 14C, į pemešimo generatorių paduodami bitas Gen, parodantis, kad suma sekcijoje yra didesnė arba lygi 16, perdavimo bitas Pr, parodantis, kad sekcijos suma yra didesnė arba lygi 15, perkėlimo bitas iš sekančios, prieš einančios, sekcijos c in i, perkėlimo bitas c in i4 iš sumavimo įtaiso išpjovos AS4 ir bitas B 1sb-i+1. Bitas B 1sb-i+1 prijungtas per invertorių 75 prie invertuoto trijų būsenų elemento įėjimo ir tiesiai į trijų būsenų elemento 76 valdymo įėjimą. Bitas c in i4 prijungtas prie antrojo invertuoto trijų būsenų elemento 76 įėjimo. Taigi, kai bitas B 1sb-i+1 yra „0“, tada trijų būsenų elementas valdomas taip, kad perduotų bitą B 1sb-i+1 į išėjimą. Kai bitas B 1sb-i+1 yra „1“, tada trijų būsenų elementas valdomas taip, kad perduotų bitą c in i4 į išėjimą, parodantį, kai c in i4 yra „1“, kad mantisė išvesta.

Mantisės perkėlimas būtinas tik tikslumo grupei, gretimai su argumento eksponentės dalimi, ir tik šiai grupei perduodamas bitas B 1sb i+1 yra lygus „1“.

Tikslumo grupė, gretima eksponentinei daliai blokuoja perkėlimus iš sekcijos į sekciją. Taigi, bitas B 1sb i+1 yra prijungtas prie elemento IR 77 invertuoto įėjimo, prie kurio antrojo invertuoto įėjimo yra prijungtas invertuotas signalas Gen. Elementas 77 perduoda signalą Gen į išėjimą tada, kai B 1sb i+1 yra „0“. Bitas B 1sb i+1, taip pat, yra prijungtas prie elemento IR 78 pirmojo invertuoto įėjimo, prie antrojo to paties elemento įėjimo prijungtas invertuotas signalas Pr, ir prie trečiojo invertuoto įėjimo bitas c in i. Elementas 78 perduoda signalą Pr į išėjimą tada, kai B 1sb i+1 yra „0“ ir c in i yra „0“. Elemento IR 79 įėjimai sujungti su ARBA elementų išėjimais ir jo išėjime formuojamas „1“, kai nors viename iš ARBA elementų išėjime yra „1“.

Sumavimo įtaiso kiekvienos išpjovos realizacija parodyta FIG. 14C. Įėjimai a ik ir b ik yra prijungti prie loginį lygiavertiškumą tikrinačių elemento 80, formuojančio „1“ išėjime, kai įėjimo signalai skiriasi. Loginį lygiavertiškumą tikrinačio elemento 80 išėjimas prijungtas prie sekančio loginį lygiavertiškumą tikrinančio elemento 81 pirmojo įėjimo, prie antrojo įėjimo - signalas (bitas) c in ik. Elemento 81 išėjime yra „1“ tiksliai tada, kai signalai įėjimuose skirtingi. Elementas 81 perduoda į išėjimą sum in ir perima perkėlimo bito, iš gretimos, prieš ją stovinčios, sumavimo įtaiso išpjovos kontrolę, arba, jeigu išpjova yra pirmoji sekcijoje, tai perkėlimo bitą iš prieš einančios sekcijos.

Bitai a ik ir b ik yra sujungti su ARBA elemento 82 įėjimais, kurio invertuotas išėjimas yra sujungtas su pirmaisiais invertuotais, kiekvieno iš trijų, IR elementų 83, 84, 85 įėjimais. Invertuotas bitas c in ik* prijungtas prie elemento IR 83 antrojo invertuoto įėjimo. Elementas 83 perduoda bitą c in ik, kai bent vienas iš a ik arba b ik yra „1“. Invertuotas bitas P in ik* iš sekančios, prieš einančios sumatoriaus išpjovos, prijungtas prie elemento IR 84 antrojo invertuoto įėjimo. Elementas 84 perduoda bitą P in ik, kai bent vienas iš a ik arba b ik yra „1“. Invertuotas bitas G in ik* prijungtas prie elemento IR 85 antrojo invertuoto įėjimo. Elementas 84 perduoda bitą G in ik, kai bent vienas iš a ik arba b ik yra „1“. Kaip parodyta FIG. 14B, bitas P in ik ir bitas G in ik yra „1“ ir „0“ kiekvienai pirmajai sumatoriaus išpjovai AS1 kiekvienoje tikslumo sekcijoje, atitinkamai. Bitai Pr ir Gen reikalingi realizuoti perkėlimą „salve“, t.y. pagreitinti sumatoriaus darbą. Perkėlimo grandinių įėjimuose

esantys diskretūs dydžiai visose tikslumo grupėse suskaičiuojami lygiagrečiai, t.y. vienu metu visoms tikslumo grupėms. Paskutinis perkėlimas c atsiranda po laiko tarpo lygiaus $4+8$ loginio elemento vėlinimams, kai įprasto pernešimo atveju - 32 loginio elemento vėlinimams. Bitai a ir b ir b ir b , taip pat, yra sujungti su IR elemento 88 įėjimu, kurio invertuotas išėjimas yra sujungtas su kiekvieno iš dviejų, ARBA elementų 86, 87 pirmaisiais įėjimais. Elemento IR 83 išėjimas yra sujungtas su elemento ARBA 86, formuojančiu išėjimo signalą c out ir, antruoju įėjimu. Elemento ARBA 85 išėjimas yra sujungtas su IR elemento 87, formuojančiu signalą c out ir, antruoju įėjimu.

Tam, kad realizuoti labai greitą perkėlimą, trunkantį tik vieną loginio elemento vėlinimo laiką, bito išėjimas invertuojamas perkėlimo generavimo signalo poliarizumas ir perduodami bitai yra invertuojami (neparodyta). Taigi, kiekviena antra tikslumo grupės sekcija turėtų invertuotus įėjimus.

Kaip matyti iš FIG. 14A kairiosios pusės, reikšmingiausia tikslumo grupė G0 nežymiai skiriasi nuo kitų sekcijų. Šioje sekcijoje nenaudojamas įėjimo argumentas B lsb-o. Jis lygus „0“ grupėje. Vietoj jo yra mantinės perkėlimo įėjimas c in m. Jis naudojamas generuoti perkėlimą į ženklą bitą. Išėjime yra papildomas bitas parodantis perkėlimą iš eksponentės. Perkėlimo signalas generuojamas kaip ir kitose grupėse, išskyrus tai, kad B lsb o lygus „0“, t.y. „neteisinga“.

Vieno bito signalai $c5$ 1, $c5$ 2, $c5$ 3 gali turėti sekančius bitų paternus: ($c5$ 1, $c5$ 2, $c5$ 3) = („neteisinga“, „neteisinga“, „neteisinga“) - reiškiantį sveiką skaičių; ($c5$ 1, $c5$ 2, $c5$ 3) = („neteisinga“, „neteisinga“, „teisinga“) - reiškiantį, kad abiejų argumentų a ir b normalizuotas kodas; ($c5$ 1, $c5$ 2, $c5$ 3) = („teisinga“, „neteisinga“, „neteisinga“) - reiškiantį, kad argumento a - keičiamas kodas, o argumento b - normalizuotas kodas; ($c5$ 1, $c5$ 2, $c5$ 3) = („teisinga“, „teisinga“, „teisinga“) - reiškiantį, kad argumento a - normalizuotas kodas, o argumento b - keičiamas kodas.

FIG. 15

Inkrementoriaus struktūra labai panaši į sumatoriaus struktūrą. FIG. 15 parodyta inkrementoriaus tikslumo sekcijos realizacija. Inkrementoriaus turi tikrai vieną įėjimą a' (neparodytas). Jo bitų išėjimas AS1' - AS4' skiriasi nuo sumatoriaus bitų išėjimų tuo, kad neturi Gen bito, perduodamo iš vienos išėjimos kitai. Invertuotas išėjimas Pr iš išėjimos AS4' ir invertuotas perkėlimas c in* iš gretimos prieš stovinčios sekcijos prijungti prie elemento ARBA 92, formuojančio perkėlimo išėjimo signalą c out į sekančią sekciją, įėjimų. Perkėlimo signalo c in, iš sekcijos G7, reikšmė „1“ parodo, kad inkrementoriaus turinio padidinimas yra atliktas. Elemento IR 90 išėjimas ir bitas B lsb i+1 yra sujungti su įėjimais elemento IR 92, išduodančio perkėlimo signalą c out į sekančią sekciją. Inkrementoriaus išėjimas yra paduodamas į kitas grandines panašiai kaip sumatoriuje. Perkėlimo „salve“ grandinė c in/c out parodo kai suma yra didesnė arba lygi 2. Perkėlimo „salve“ grandinė Pe' parodo kai suma yra didesnė arba lygi 1.

Kaip ir sumatoriuje, inkrementoriuje yra mantinės perkėlimo linija cm , valdoma signalo B lsb i+1. Valdymo bitai $c1g$ ir $c2g$ parodo koks, sveikas ar slankaus kablelio, dydis yra pateiktas. Bitai $c1g$ ir $c2g$ valdo trijų būsenų loginį elementą, kuris nustato mantinės perkėlimo magistralę į užduotą būseną (neparodyta).

Dekrementoriaus panašus į inkrementorių, tačiau įėjimas į dekrementoriaus išėjimą a' in*, atitinkantis inkrementoriaus išėjimą AS1' - AS4', yra invertuotas.

Selektoriai operando įtaisuose reliuzuoti kaip loginių perdavimo elementų eilutė. Selektoriaus išėjimas visada yra apibrėžtas. Jame yra valdymo žodžių ir įėjimo dydžių kaip argumentų sąrašas. Visi, išskyrus vieną yra išjungtoje būsenoje. Kiekvienas įėjimas valdomas papildomu signalu. Pilnas selektorius yra valdomas elementų sąrašu. Jie valdomi paimti informaciją iš nustatytų portų. Pagrindinio selektoriaus bitų išėjimas parodyta FIG. 16. Kiekvieno įėjimo porto vienas iš bitų į selektorių S ina, S inb, S inc ir S ind yra prijungti prie valdomų jungiklių S a, S b, S c ir S d, atitinkamai. Valdymo pora, įgalinanti perduoti įėjimo signalą per atitinkamą jungiklį, gauna signalą „1, 0“, kitos gauna signalus „0“.

Reikia pabrėžti, kad aprašyta dauguma pagrindinių ALI grandinių, likusias, neaprašytas grandines, specialistai naudodamiesi pateiktais principais gali nesunkiai realizuoti. Be to, buvo manoma, kad nėra būtinybės aprašinėti kiekvieną grandinę smulkiau.

ALI VALDYMO GRANDINĖS APRAŠYMAS

Priede 1 parodytas ryšys tarp įėjimo ir išėjimo signalų slankaus kablelio dydžio eksponentių išlyginimo operacijoje.

Priedas suskirstytas į skirtingus valdymo variantus. Kiekvienas valdymo variantas atitinka vieną išėjimą, t.y. vieną mikrokomandą. ALI valdymo grandinės 27 išėjimai, išvardinti po pavadinimu „Mikrokomandų linijos“, apibūdinami signalų lygiais magistralėse $c1$ - $c5$. Magistralės $c1$, $c2$, $c3$ yra suskirstytos į mažesnes magistrales $c1$ 1, $c1$ 2, $c1$ 3 ir t.t., prijungtas prie įvairių ALI taškų. Skirtingos įėjimų sekos gali duoti vienodą išėjimą. Alternatyvūs įėjimo signalai, iššaukiantys vienodą išėjimą, išvardinti kaip sąlyga 1, sąlyga 2 ir t.t. Taigi, įėjimai, išvardinti po užrašu „valdymo variantas 1“ ir „būsena, linijų sąlygos 4“, iššaukia išėjimo signalus kurie pateikti dalyje „Mikrokomandų linijos“. Išėjimai, išvardyti kaip sąlygos 1, 2, ir 3 duoda vienodus išėjimus kaip sąlyga 4.

Kiekviena sąlyga yra apibrėžta tam tikra signalo lygių kombinacija (0/1) kelių būsenų linijose, iki kitoje būsenoje neįgaus neapibrėžto dydžio (X =neperduodama)

Visi signalai išvardinti prieduose yra sužymėti kaip FIG. 6, išskyrus signalus co_limit1 ir co_limit2 .

Fig. 6 signalas co_limit1 suskirstytas į dvi komponentes Bco_max1 ir Bco_min1 .

Signalas co_limit2 suskirstytas į dvi komponentes Bco_max2 ir Bco_min2 .

Išėjimas A_{STATE} yra visų grandinės 27 įėjimų kopija. Išėjimas A_{STATE} yra sujungtas su išoriniu valdymo įrenginiu VI.

Priede 2 parodytas ryšys tarp ALĮ valdymo grandinės įėjimų ir išėjimų, kai vykdoma sveikų dydžių daugyba.

ALĮ valdymo grandinė 27, parodyta FIG. 6, yra labai sudėtingos konfigūracijos programuojamų loginių elementų matrica. Matricos realizacija neparodyta, nes jos konfigūracija apskaičiuojama kompiuteriu, aprūpintu visais, naudojamais grandinėje 27, algoritmais, bei valdymo ir informaciniais signalais. Loginių elementų matrica gaunama kaip mikroschemos maskavimo paternai, naudojant kompiuterį, su jau suderinta visam ALĮ darbui maskavimo paternų mikroschemos konfigūracija. Taigi, šios grandinės neįmanoma detaliau pavaizduoti. ALĮ valdymo grandinės 27 įėjimo ir išėjimo signalų sąryšis, kai atliekama sveikų skaičių daugyba ir eksponentės dalių išlyginimas slankaus kabelio skaičių sudėtyje, parodytas prieduose.

Kol išradimas buvo aprašomas, remiantis tuo metu žinomais elementais, galėjo pasikeisti ir papildyti jų bazė, kurioje buvo realizuotas išradimas. Galimos realizacijos modifikacijos, naudojant naują elementų bazę, nenukrypstant nuo išradimo dvasios ir esmės.

Dydis paduodamas iš tikslo registro arba G-registro arba iš kito vidinio ALĮ įrenginio kaip komandos operandas/daugyba-santykis arba polinomas. Informacija, nustatanti atspindėjimo tipą likusioje bitų eilutėje magistralėje gali būti paduodama komandų magistrale arba į kodo lauką.

KONTROLINIAI UŽDAVINIAI

EKSPONENČIŲ IŠLYGINIMAS SUDEDANT SLANKAUS KABELELIO SKAIČIUS

kontrolinis pavyzdys 1

būsena, reikšmingos linijos-sąlyga 1

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, 1)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 1, X, X)
sign2         (1, 0, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         1
co_limit1     (X, 0)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būsena, reikšmingos linijos-sąlyga 2

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, 1)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 1, X, X)
sign2         (0, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         1
co_limit1     (X, 0)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būsena, reikšmingos linijos-sąlyga 3

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_expl2     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 1, X, X)
sign2         (1, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_eil         X
gr_d1         (X, X, X)
c_ed1         1
co_limit1     (X, 0)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būsenā, reikšmingos linijos- sąlyga 4

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_expl2     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 1, X, X)
sign2         (0, 0, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_eil         X
gr_d1         (X, X, X)
c_ed1         1
co_limit1     (X, 0)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

MIKROINSTRUKCIJŲ LINIJOS

```

c1
c1_1          (0, 1, 0, 0, 0, 0)
c1_2          [(0, 1); (0, 1); (1, 0); (0, 1)]
c1_3          [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
c1_4          [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
c1_5          [(1, 0); (0, 1); (0, 1)]
c1_6          [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c1_7          (0, [(0, 1); (1, 0)], [0; 0])

```

c1_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c1_10 [0; 0; 0; 0; 1]

 c2
 c2_1 (0, 0, 0, 1, 0, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ([(1, 0); (0, 1)], 0)

 c4 (0, 1)

 c5 (0, 0, 1)

kontrolinis pavyzdys 2

būseną, reikšmingos linijos- sąlyga 1

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, 1)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 0, X, X)
sign2         (1, 0, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         1
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, 0)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būseną, reikšmingos linijos- sąlyga 2

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, 1)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 0, X, X)
sign2         (0, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         1
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, 0)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būseną, reikšmingos linijos- sąlyga 3

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)

```

```

cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (1, 0, X, X)
sign2      (1, 1, X, X)
sign3      (X, X, X, X)
gr_il      (X, X, X)
c_ei1      1
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, 0)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 4

```

cmp_prec   X
cmp01      (X, X)
cmp12      (X, X)
cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (1, 0, X, X)
sign2      (0, 0, X, X)
sign3      (X, X, X, X)
gr_il      (X, X, X)
c_ei1      1
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, 0)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

cl
cl_1       (0, 1, 0, 0, 0, 0)
cl_2       [(0, 1); (0, 1); (1, 0); (0, 1)]
cl_3       [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1)]
cl_4       [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
cl_5       [(1, 0); (0, 1); (0, 1)]
cl_6       [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
cl_7       (0, [(0, 1); (1, 0)], [1; 1])
cl_8       (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
cl_10      [0; 0; 0; 0; 1]

```

c2
 c2_1 (0, 0, 0, 1, 0, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ((1, 0); (0, 1)), 0)

 c4 (0, 1)

 c5 (0, 0, 1)

c2
 c2_1 (0, 0, 0, 1, 0, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ([(1, 0); (0, 1)], 0)

 c4 (0, 1)

 c5 (0, 0, 1)

kontrolinis pavyzdys 3

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, 1)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 1, X, X)
sign2         (1, 0, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         1
co_limit1     (X, 0)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būsena, reikšmingos linijos - sąlyga 2

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, 1)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 1, X, X)
sign2         (0, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         1
co_limit1     (X, 0)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 3

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)

```

```

cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (0, 1, X, X)
sign2      (1, 1, X, X)
sign3      (X, X, X, X)
gr_il      (X, X, X)
c_eil      X
gr_dl      (X, X, X)
c_edl      1
co_limit1  (X, 0)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 4

```

cmp_prec   X
cmp01      (X, X)
cmp12      (X, X)
cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (0, 1, X, X)
sign2      (0, 0, X, X)
sign3      (X, X, X, X)
gr_il      (X, X, X)
c_eil      X
gr_dl      (X, X, X)
c_edl      1
co_limit1  (X, 0)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

mikroinstrukcijų linijos

```

cl
cl_1       (0, 1, 0, 0, 0, 0)
cl_2       [(0, 1); (0, 1); (1, 0); (0, 1)]
cl_3       [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
cl_4       [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
cl_5       [(1, 0); (0, 1); (0, 1)]
cl_6       [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
cl_7       (0, [(0, 1); (1, 0)], [1; 1])
cl_8       (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
cl_10      [0; 0; 0; 0; 1]

```

```

c2
c2_1      (0, 0, 0, 1, 0, 0)
c2_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c2_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c2_4      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c2_5      [(0, 1); (1, 0); (0, 1)]
c2_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_7      (0, [(0, 1); (1, 0)], [0; 0])
c2_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c2_10     [0; 0; 0; 0; 1]

c3
c3_1      (1, 0)
c3_2      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c3_3      (0, [(0, 1); (1, 0)], [0; 0])
c3_4      [0; 0; 0; 0; 1]
c3_5      ([ (1, 0); (0, 1) ], 0)

c4
c4         (0, 1)

c5
c5         (0, 0, 1)

```

kontrolinis pavyzdys 4

būseną, reikšmingos linijos- sąlyga 1

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, 1)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 0, X, X)
sign2         (1, 0, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         1
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, 0)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būseną, reikšmingos linijos- sąlyga 2

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, 1)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 0, X, X)
sign2         (0, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         1
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, 0)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būseną, reikšmingos linijos- sąlyga 3

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)

```

cmp23 (X, X)
 cmp_exp12 (X, X)
 insig1 ((X, X), (X, X), (X, X))
 insig2 ((X, X), (X, X), (X, X))
 insig3 ((X, X), (X, X), (X, X))
 sign1 (1, 0, X, X)
 sign2 (1, 1, X, X)
 sign3 (X, X, X, X)
 gr_i1 (X, X, X)
 c_ei1 1
 gr_d1 (X, X, X)
 c_ed1 X
 co_limit1 (X, 0)
 gr_i2 (X, X, X)
 c_ei2 X
 gr_d2 (X, X, X)
 c_ed2 X
 co_limit2 (X, X)
 gr_a1 (X, X, X)
 gr_a2 (X, X, X)
 gr_a3 (X, X, X)
 ins [0; 0; 0; 0; 1; 1; 0; 0; 1]

būsena, reikšmingos linijos - sąlyga 4

cmp_prec X
 cmp01 (X, X)
 cmp12 (X, X)
 cmp23 (X, X)
 cmp_exp12 (X, X)
 insig1 ((X, X), (X, X), (X, X))
 insig2 ((X, X), (X, X), (X, X))
 insig3 ((X, X), (X, X), (X, X))
 sign1 (1, 0, X, X)
 sign2 (0, 0, X, X)
 sign3 (X, X, X, X)
 gr_i1 (X, X, X)
 c_ei1 1
 gr_d1 (X, X, X)
 c_ed1 X
 co_limit1 (X, 0)
 gr_i2 (X, X, X)
 c_ei2 X
 gr_d2 (X, X, X)
 c_ed2 X
 co_limit2 (X, X)
 gr_a1 (X, X, X)
 gr_a2 (X, X, X)
 gr_a3 (X, X, X)
 ins [0; 0; 0; 0; 1; 1; 0; 0; 1]

mikroinstrukcijų linijos

cl
 cl_1 (0, 1, 0, 0, 0, 0)
 cl_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 cl_3 [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1)]
 cl_4 [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
 cl_5 [(1, 0); (0, 1); (0, 1)]
 cl_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 cl_7 (0, [(0, 1); (1, 0)], [0; 0])
 cl_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 cl_10 [0; 0; 0; 0; 1]

c2
 c2_1 (0, 0, 0, 1, 0, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ([(1, 0); (0, 1)], 0)

 c4 (0, 1)

 c5 (0, 0, 1)

kontrolinis pavyzdys 5

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 0, X, X)
sign2         (0, 0, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         1
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būsena, reikšmingos linijos - sąlyga 2

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 0, X, X)
sign2         (1, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         1
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

cl
cl_1          (0, 0, 1, 0, 0, 1)
cl_2          [(0, 1); (0, 1); (1, 0); (0, 1)]

```

c1_3 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
 c1_4 [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
 c1_5 [(0, 1); (0, 1); (1, 0)]
 c1_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c1_7 (0, [(0, 1); (1, 0)], [0; 0])
 c1_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c1_10 [0; 0; 0; 0; 1]

 c2
 c2_1 (0, 0, 0, 1, 0, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ([(1, 0); (0, 1)], 0)

 c4 (0, 1)

 c5 (0, 0, 1)

būsena, reikšmingos linijos- sąlyga 1

```
cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 1, X, X)
sign2         (0, 0, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         1
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]
```

būsena, reikšmingos linijos-sąlyga 2

```
cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 1, X, X)
sign2         (1, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         1
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; -0; 0; 1; 0; 0; 0; 1]
```

mikroinstrukcijų linijos

```
cl
cl_1          (0, 0, 1, 0, 0, 0)
cl_2          [(0, 1); (0, 1); (1, 0); (0, 1)]
```

c1_3 [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1)]
 c1_4 [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
 c1_5 [(0, 1); (0, 1); (1, 0)]
 c1_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c1_7 (0, [(0, 1); (1, 0)], [1; 1])
 c1_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c1_10 [0; 0; 0; 0; 1]

 c2
 c2_1 (0, 0, 0, 1, 0, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ([(1, 0); (0, 1)], 0)

 c4 (0, 1)
 c5 (0, 0, 1)

kontrolinis pavyzdys 7

būsena, reikšmingos linijos-sąlyga 1

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 0, X, X)
sign2         (0, 0, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         1
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 2

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 0, X, X)
sign2         (1, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         1
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

mikroinstrukcijų linijos

```

cl
cl_1          (0, 0, 1, 0, 0, 1)
cl_2          [(0, 1); (0, 1); (1, 0); (0, 1)]

```

c1_3 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
 c1_4 [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
 c1_5 [(0, 1); (0, 1); (1, 0)]
 c1_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c1_7 (0, [(0, 1); (1, 0)], [1; 1])
 c1_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c1_10 [0; 0; 0; 0; 1]

 c2
 c2_1 (0, 0, 0, 1, 0, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ([1, 0]; (0, 1)), 0

 c4 (0, 1)

 c5 (0, 0, 1)

kontrolinis pavyzdys 8

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 1, X, X)
sign2         (0, 0, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         1
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 2

```

cmp_prec      0
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 1, X, X)
sign2         (1, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         1
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

mikroinstrukcijų linijos

```

c1
c1_1          (0, 0, 1, 0, 0, 0)
c1_2          [(0, 1); (0, 1); (1, 0); (0, 1)]

```

c1_3 [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1)]
 c1_4 [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
 c1_5 [(0, 1); (0, 1); (1, 0)]
 c1_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c1_7 (0, [(0, 1); (1, 0)], [0; 0])
 c1_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c1_10 [0; 0; 0; 0; 1]

 c2
 c2_1 (0, 0, 0, 1, 0, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ([(1, 0); (0, 1)], 0)

 c4 (0, 1)

 c5 (0, 0, 1)

kontrolinis pavyzdys 9

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 0, X, X)
sign2         (0, 0, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 2

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 0, X, X)
sign2         (1, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 3

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)

```

```

cmp23      (X, X)
cmp_exp12  (0, 0)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (0, 1, X, X)
sign2      (0, 1, X, X)
sign3      (X, X, X, X)
gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būsena, rekšmingos linijos- sąlyga 4

```

cmp_prec   1
cmp01      (X, X)
cmp12      (X, X)
cmp23      (X, X)
cmp_exp12  (0, 0)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (0, 1, X, X)
sign2      (1, 0, X, X)
sign3      (X, X, X, X)
gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

cl         (0, 1, 0, 0, 0, 0)
cl_1       [(0, 1); (0, 1); (1, 0); (0, 1)]
cl_2       [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
cl_3       [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
cl_4       [(0, 1); (1, 0); (0, 1)]
cl_5       [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
cl_6       (0, [(0, 1); (1, 0)], [0; 0])
cl_7       (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
cl_8       [0; 0; 0; 0; 1]
cl_10

```

43

```

c2
c2_1      (0, 0, 0, 1, 0, 0)
c2_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c2_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c2_4      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c2_5      [(0, 1); (1, 0); (0, 1)]
c2_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_7      (0, [(0, 1); (1, 0)], [0; 0])
c2_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c2_10     [0; 0; 0; 0; 1]

c3
c3_1      (1, 0)
c3_2      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c3_3      (0, [(0, 1); (1, 0)], [0; 0])
c3_4      [0; 0; 0; 0; 1]
c3_5      ([ (1, 0); (0, 1) ], 0)

c4        (0, 1)

c5        (0, 0, 1)

```

kontrolinis pavyzdys 10

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 0, X, X)
sign2         (0, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 2

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 0, X, X)
sign2         (1, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 0; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 3

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)

```

cmp23 (X, X)
 cmp_exp12 (0, 0)
 insig1 ((X, X), (X, X), (X, X))
 insig2 ((X, X), (X, X), (X, X))
 insig3 ((X, X), (X, X), (X, X))
 sign1 (1, 1, X, X)
 sign2 (0, 0, X, X)
 sign3 (X, X, X, X)
 gr_i1 (X, X, X)
 c_ei1 X
 gr_d1 (X, X, X)
 c_ed1 X
 co_limit1 (X, X)
 gr_i2 (X, X, X)
 c_ei2 X
 gr_d2 (X, X, X)
 c_ed2 X
 co_limit2 (X, X)
 gr_a1 (X, X, X)
 gr_a2 (X, X, X)
 gr_a3 (X, X, X)
 ins [0; 0; 0; 0; 1; 0; 0; 0; 1]

būsena, rekšmingos lūnijos- sąlyga 4

cmp_prec 1
 cmp01 (X, X)
 cmp12 (X, X)
 cmp23 (X, X)
 cmp_exp12 (0, 0)
 insig1 ((X, X), (X, X), (X, X))
 insig2 ((X, X), (X, X), (X, X))
 insig3 ((X, X), (X, X), (X, X))
 sign1 (1, 1, X, X)
 sign2 (1, 1, X, X)
 sign3 (X, X, X, X)
 gr_i1 (X, X, X)
 c_ei1 X
 gr_d1 (X, X, X)
 c_ed1 X
 co_limit1 (X, X)
 gr_i2 (X, X, X)
 c_ei2 X
 gr_d2 (X, X, X)
 c_ed2 X
 co_limit2 (X, X)
 gr_a1 (X, X, X)
 gr_a2 (X, X, X)
 gr_a3 (X, X, X)
 ins [0; 0; 0; 0; 1; 0; 0; 0; 1]

mikroinstrukcijų linijos

cl
 cl_1 (0, 1, 0, 0, 0, 0)
 cl_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 cl_3 [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1)]
 cl_4 [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
 cl_5 [(0, 1); (1, 0); (0, 1)]
 cl_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 cl_7 (0, [(0, 1); (1, 0)], [1; 1])
 cl_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 cl_10 [0; 0; 0; 0; 1]

c2
 c2_1 (0, 0, 0, 1, 0, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ([(1, 0); (0, 1)], 0)

 c4 (0, 1)

 c5 (0, 0, 1)

kontrolinis pavyzdys 11

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 0, X, X)
sign2         (0, 0, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 2

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, 0, X, X)
sign2         (1, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būsena, reikšmingos linijos - sąlyga 3

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)

```

48

```

cmp23      (X, X)
cmp_exp12  (0, 0)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (0, 1, X, X)
sign2      (0, 1, X, X)
sign3      (X, X, X, X)
gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būsena, rekšmingos linijos-sąlyga 2

```

cmp_prec   1
cmp01      (X, X)
cmp12      (X, X)
cmp23      (X, X)
cmp_exp12  (0, 0)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (0, 1, X, X)
sign2      (1, 0, X, X)
sign3      (X, X, X, X)
gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

mikroinstrukcijų linijos

```

cl
cl_1       (0, 1, 0, 0, 0, 0)
cl_2       [(0, 1); (0, 1); (1, 0); (0, 1)]
cl_3       [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
cl_4       [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
cl_5       [(0, 1); (1, 0); (0, 1)]
cl_6       [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
cl_7       (0, [(0, 1); (1, 0)], [1; 1])
cl_8       (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
cl_10      [0; 0; 0; 0; 1]

```

```

c2
c2_1      (0, 0, 0, 1, 0, 0)
c2_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c2_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c2_4      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c2_5      [(0, 1); (1, 0); (0, 1)]
c2_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_7      (0, [(0, 1); (1, 0)], [0; 0])
c2_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c2_10     [0; 0; 0; 0; 1]

c3
c3_1      (1, 0)
c3_2      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c3_3      (0, [(0, 1); (1, 0)], [0; 0])
c3_4      [0; 0; 0; 0; 1]
c3_5      ([[(1, 0); (0, 1)], 0)

c4        (0, 1)

c5        (0, 0, 1)

```

kontrolinis pavyzdys 12

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 0, X, X)
sign2         (0, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 2

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (0, 0)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, 0, X, X)
sign2         (1, 1, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 3

```

cmp_prec      1
cmp01         (X, X)
cmp12         (X, X)

```

```

cmp23      (X, X)
cmp_exp12  (0, 0)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (1, 1, X, X)
sign2      (0, 0, X, X)
sign3      (X, X, X, X)
gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

būsena, rekšmingos linijos- sąlyga 4

```

cmp_prec   1
cmp01      (X, X)
cmp12      (X, X)
cmp23      (X, X)
cmp_exp12  (0, 0)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (1, 1, X, X)
sign2      (1, 1, X, X)
sign3      (X, X, X, X)
gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [0; 0; 0; 0; 1; 1; 0; 0; 1]

```

mikroinstrukcijų linijos

```

c1
c1_1       (0, 1, 0, 0, 0, 0)
c1_2       [(0, 1); (0, 1); (1, 0); (0, 1)]
c1_3       [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1)]
c1_4       [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
c1_5       [(0, 1); (1, 0); (0, 1)]
c1_6       [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c1_7       (0, [(0, 1); (1, 0)], [0; 0])
c1_8       (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c1_10      [0; 0; 0; 0; 1]

```

```

c2
c2_1      (0, 0, 0, 1, 0, 0)
c2_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c2_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c2_4      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c2_5      [(0, 1); (1, 0); (0, 1)]
c2_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_7      (0, [(0, 1); (1, 0)], [0; 0])
c2_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c2_10     [0; 0; 0; 0; 1]

c3
c3_1      (1, 0)
c3_2      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c3_3      (0, [(0, 1); (1, 0)], [0; 0])
c3_4      [0; 0; 0; 0; 1]
c3_5      ([ (1, 0); (0, 1) ], 0)

c4        (0, 1)

c5        (0, 0, 1)

```

SVEIKO SKAIČIAUS DAUGYBA

kontrolinis pavyzdys 1

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_expl2     (X, X)
insig1        ((X, X), (1, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (1, X, X, X)
sign2         (X, X, X, X)
sign3         (X, X, X, X)
gr_il         (X, X, X)
c_eil         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 0; 0; 0; 0; 0; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 2

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_expl2     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (1, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (X, X, X, X)
sign2         (1, X, X, X)
sign3         (X, X, X, X)
gr_il         (X, X, X)
c_eil         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 0; 0; 0; 0; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

c1
c1_1      (0, 0, 0, 0, 1, 0)
c1_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c1_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c1_4      [(0, 1); (0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c1_5      [(0, 1); (1, 0); (0, 1)]
c1_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c1_7      (0, [(0, 1); (1, 0)], [0; 0])
c1_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c1_10     [0; 0; 0; 0; 1]

c2
c2_1      (0, 0, 0, 1, 0, 0)
c2_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c2_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c2_4      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c2_5      [(0, 1); (1, 0); (0, 1)]
c2_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_7      (0, [(0, 1); (1, 0)], [0; 0])
c2_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c2_10     [0; 0; 0; 0; 1]

c3
c3_1      (1, 0)
c3_2      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c3_3      (0, [(0, 1); (1, 0)], [0; 0])
c3_4      [0; 0; 0; 0; 1]
c3_5      ([ (1, 0); (0, 1) ], 0)

c4
c4         (0, 1)

c5
c5         (0, 0, 0)

```

kontrolinis pavyzdys 2

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_expl2     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, X, X, X)
sign2         (0, X, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_eil         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 0; 0; 0; 0; 0; 0; 0; 1]

```

būsena, reikšmingos - sąlyga 2

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_expl2     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (0, X), (X, X))
insig3        ((X, X), (X, X), (X, X))
sign1         (0, X, X, X)
sign2         (X, X, X, X)
sign3         (X, X, X, X)
gr_i1         (X, X, X)
c_eil         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 0; 0; 0; 0; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)

```

```

cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (0, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (X, X, X, X)
sign2      (0, X, X, X)
sign3      (X, X, X, X)
gr_il      (X, X, X)
c_eil      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [1; 0; 0; 0; 0; 0; 0; 0; 1]

```

būsenā, reikšmingos linijos- sąlyga 4

```

cmp_prec   X
cmp01      (X, X)
cmp12      (X, X)
cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (0, X), (X, X))
insig2     ((X, X), (0, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (X, X, X, X)
sign2      (X, X, X, X)
sign3      (X, X, X, X)
gr_il      (X, X, X)
c_eil      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [1; 0; 0; 0; 0; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

cl
cl_1       (0, 0, 0, 0, 1, 0)
cl_2       [(0, 1); (0, 1); (1, 0); (0, 1)]
cl_3       [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
cl_4       [(0, 1); (0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
cl_5       [(0, 1); (1, 0); (0, 1)]
cl_6       [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
cl_7       (0, [(0, 1); (1, 0)], [0; 0])
cl_8       (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
cl_10      [0; 0; 0; 0; 1]

```

c2
 c2_1 (0, 0, 0, 0, 1, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ([(1, 0); (0, 1)], 0)

 c4 (0, 1)

 c5 (0, 0, 0)

kontrolinis pavyzdys 3

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (1, X), (X, X))
sign1         (X, X, X, X)
sign2         (X, X, X, X)
sign3         (1, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 2

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (1, X), (X, X))
sign1         (X, X, X, X)
sign2         (X, X, X, X)
sign3         (1, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 0; 1; 0; 0; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

cl
cl_1          (0, 0, 0, 1, 0, 0)
cl_2          [(0, 1); (0, 1); (1, 0); (0, 1)]

```

c1_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c1_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c1_5 [(0, 1); (1, 0); (0, 1)]
 c1_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c1_7 (0, [(0, 1); (1, 0)], [0; 0])
 c1_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c1_10 [0; 0; 0; 0; 1]

 c2
 c2_1 (0, 0, 0, 1, 0, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ([(1, 0); (0, 1)], 0)

 c4 (0, 1)

 c5 (0, 0, 0)

kontrolinis pavyzdys 4

būsena, reikšmingos linijos

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_expl2     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (0, X), (X, X))
sign1         (X, X, X, X)
sign2         (X, X, X, X)
sign3         (1, X, X, 0)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

c1
c1_1          (0, 0, 0, 1, 0, 0)
c1_2          [(0, 1); (0, 1); (1, 0); (0, 1)]
c1_3          [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c1_4          [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c1_5          [(0, 1); (1, 0); (0, 1)]
c1_6          [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c1_7          (0, [(0, 1); (1, 0)], [0; 0])
c1_8          (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c1_10         [0; 0; 0; 0; 1]

c2
c2_1          (0, 0, 0, 0, 1, 0)
c2_2          [(0, 1); (0, 1); (1, 0); (0, 1)]
c2_3          [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c2_4          [(0, 1); (1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_5          [(0, 1); (1, 0); (0, 1)]
c2_6          [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_7          (0, [(0, 1); (1, 0)], [0; 0])
c2_8          (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c2_10         [0; 0; 0; 0; 1]

c3
c3_1          (1, 0)
c3_2          [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
c3_3          (0, [(0, 1); (1, 0)], [0; 0])
c3_4          [0; 0; 0; 0; 1]
c3_5          [(1, 0); (0, 1), 0)

c4            (0, 1)

c5            (0, 0, 0)

```

kontrolinis pavyzdys 4

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (0, X), (X, X))
sign1         (X, X, X, X)
sign2         (1, X, X, X)
sign3         (1, X, X, 1)
gr_i1         (X, X, X)
c_eil        X
gr_d1         (X, X, X)
c_ed1        X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2        X
gr_d2         (X, X, X)
c_ed2        X
co_limit2     (X, X)
gr_a1         (X, 1, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 2

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, 1), (X, X))
sign1         (X, X, X, X)
sign2         (0, X, X, X)
sign3         (0, X, X, X)
gr_i1         (X, X, X)
c_eil        X
gr_d1         (X, X, X)
c_ed1        X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2        X
gr_d2         (X, X, X)
c_ed2        X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, 1, X)
gr_a3         (X, X, X)

ins           [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 3

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)

```

```

cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, 0), (X, X))
sign1      (X, X, X, X)
sign2      (0, X, X, X)
sign3      (0, X, X, 0)
gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, 1, X)
gr_a3      (X, X, X)

ins        [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

būšana, reikšmingos linijos- sąlyga 4

```

cmp_prec   X
cmp01      (X, X)
cmp12      (X, X)
cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (0, X), (X, X))
sign1      (X, X, X, X)
sign2      (1, X, X, X)
sign3      (1, X, X, X)
gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [1; 0; 1; 0; 0; 0; 0; 0; 1]

```

būšana, reikšmingos linijos- sąlyga 5

```

cmp_prec   X
cmp01      (X, X)
cmp12      (X, X)
cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (X, X, X, X)
sign2      (0, X, X, X)
sign3      (0, X, X, X)

```

```

gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

```

```
ins        [1; 0; 1; 0; 0; 0; 0; 0; 1]
```

mikroinstrukcijų linijos

```

c1
c1_1      (0, 0, 0, 0, 1, 0)
c1_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c1_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c1_4      [(0, 1); (0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c1_5      [(0, 1); (1, 0); (0, 1)]
c1_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c1_7      (0, [(0, 1); (1, 0)], [0; 0])
c1_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c1_10     [1; 1; 1; 1; 1]

```

```

c2
c2_1      (0, 0, 0, 1, 0, 0)
c2_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c2_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c2_4      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c2_5      [(0, 1); (1, 0); (0, 1)]
c2_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_7      (0, [(0, 1); (1, 0)], [0; 0])
c2_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c2_10     [0; 0; 0; 0; 1]

```

```

c3
c3_1      (1, 0)
c3_2      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c3_3      (0, [(0, 1); (1, 0)], [0; 0])
c3_4      [0; 0; 0; 0; 1]
c3_5      ([ (1, 0); (0, 1) ], 0)

```

```
c4        (0, 1)
```

```
c5        (0, 0, 0)
```

kontrolinis pavyzdys 6

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (0, X), (X, X))
sign1         (X, X, X, X)
sign2         (0, X, X, X)
sign3         (1, X, X, 1)
gr_il         (X, X, X)
c_eil         X
gr_dl         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, 0, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 2

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, 1), (X, X))
sign1         (X, X, X, X)
sign2         (1, X, X, X)
sign3         (0, X, X, X)
gr_il         (X, X, X)
c_eil         X
gr_dl         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, 0, X)
gr_a3         (X, X, X)

ins           [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

būsena, reikšmingos linijos - sąlyga 3

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)

```

```

cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, 0), (X, X))
sign1      (X, X, X, X)
sign2      (1, X, X, X)
sign3      (0, X, X, 0)
gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, 0, X)
gr_a3      (X, X, X)

ins        [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

būsenā, reikšmīngos līnijas - sālyga 4

```

cmp_prec   X
cmp01      (X, X)
cmp12      (X, X)
cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (0, X), (X, X))
sign1      (X, X, X, X)
sign2      (0, X, X, X)
sign3      (1, X, X, X)
gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [1; 0; 1; 0; 0; 0; 0; 0; 1]

```

būsenā, reikšmīngos līnijas - sālyga 5

```

cmp_prec   X
cmp01      (X, X)
cmp12      (X, X)
cmp23      (X, X)
cmp_exp12  (X, X)
insig1     ((X, X), (X, X), (X, X))
insig2     ((X, X), (X, X), (X, X))
insig3     ((X, X), (X, X), (X, X))
sign1      (X, X, X, X)
sign2      (1, X, X, X)
sign3      (0, X, X, X)

```

```

gr_i1      (X, X, X)
c_ei1      X
gr_d1      (X, X, X)
c_ed1      X
co_limit1  (X, X)
gr_i2      (X, X, X)
c_ei2      X
gr_d2      (X, X, X)
c_ed2      X
co_limit2  (X, X)
gr_a1      (X, X, X)
gr_a2      (X, X, X)
gr_a3      (X, X, X)

ins        [1; 0; 1; 0; 0; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

c1
c1_1      (0, 0, 0, 0, 1, 0)
c1_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c1_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c1_4      [(0, 1); (0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c1_5      [(0, 1); (1, 0); (0, 1)]
c1_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c1_7      (0, [(0, 1); (1, 0)], [0; 0])
c1_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c1_10     [0; 0; 0; 0; 0]

c2
c2_1      (0, 0, 0, 1, 0, 0)
c2_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c2_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c2_4      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c2_5      [(0, 1); (1, 0); (0, 1)]
c2_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_7      (0, [(0, 1); (1, 0)], [0; 0])
c2_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c2_10     [0; 0; 0; 0; 1]

c3
c3_1      (1, 0)
c3_2      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c3_3      (0, [(0, 1); (1, 0)], [0; 0])
c3_4      [0; 0; 0; 0; 1]
c3_5      [(1, 0); (0, 1), 0]

c4        (0, 1)

c5        (0, 0, 0)

```

kontrolinis pavyzdys 7

būsena, reikšmingos linijos - sąlyga 1

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (0, X), (X, X))
sign1         (X, X, X, X)
sign2         (0, X, X, X)
sign3         (1, X, X, 1)
gr_i1         (X, X, X)
c_eil         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, 1, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

būsena, reikšmingos linijos - sąlyga 2

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (0, X), (X, X))
sign1         (X, X, X, X)
sign2         (1, X, X, X)
sign3         (1, X, X, 1)
gr_i1         (X, X, X)
c_eil         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, 0, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

cl
cl_1          (0, 0, 0, 0, 1, 0)
cl_2          [(0, 1); (0, 1); (1, 0); (0, 1)]

```

c1_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c1_4 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
 c1_5 [(0, 1); (1, 0); (0, 1)]
 c1_6 [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
 c1_7 (0, [(0, 1); (1, 0)], [0; 0])
 c1_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c1_10 [0; 0; 0; 0; 1]

 c2
 c2_1 (0, 0, 0, 0, 1, 0)
 c2_2 [(0, 1); (0, 1); (1, 0); (0, 1)]
 c2_3 [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
 c2_4 [(0, 1); (1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_5 [(0, 1); (1, 0); (0, 1)]
 c2_6 [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
 c2_7 (0, [(0, 1); (1, 0)], [0; 0])
 c2_8 (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
 c2_10 [0; 0; 0; 0; 1]

 c3
 c3_1 (1, 0)
 c3_2 [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
 c3_3 (0, [(0, 1); (1, 0)], [0; 0])
 c3_4 [0; 0; 0; 0; 1]
 c3_5 ([(1, 0); (0, 1)], 0)

 c4 (0, 1)

 c5 (0, 0, 0)

kontrolinis pavyzdys 8

būsena, reikšmingos linijos- sąlyga 1

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, 1), (X, X))
sign1         (X, X, X, X)
sign2         (0, X, X, X)
sign3         (0, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, 0, X)
gr_a3         (X, X, X)

ins           [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

būsena, reikšmingos linijos- sąlyga 2

```

cmp_prec      X
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, 1), (X, X))
sign1         (X, X, X, X)
sign2         (1, X, X, X)
sign3         (0, X, X, X)
gr_i1         (X, X, X)
c_ei1         X
gr_d1         (X, X, X)
c_ed1         X
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         X
gr_d2         (X, X, X)
c_ed2         X
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, 1, X)
gr_a3         (X, X, X)

ins           [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

cl
cl_1          (0, 0, 0, 0, 1, 0)
cl_2          ((0, 1); (0, 1); (1, 0); (0, 1))

```

70

```

c1_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c1_4      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c1_5      [(0, 1); (1, 0); (0, 1)]
c1_6      [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1)]
c1_7      (0, [(0, 1); (1, 0)], [0; 0])
c1_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c1_10     [0; 0; 0; 0; 1]

c2
c2_1      (0, 0, 0, 1, 0, 0)
c2_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c2_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c2_4      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c2_5      [(0, 1); (1, 0); (0, 1)]
c2_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_7      (0, [(0, 1); (1, 0)], [0; 0])
c2_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c2_10     [0; 0; 0; 0; 1]

c3
c3_1      (1, 0)
c3_2      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c3_3      (0, [(0, 1); (1, 0)], [0; 0])
c3_4      [0; 0; 0; 0; 1]
c3_5      ({(1, 0); (0, 1)}, 0)

c4        (0, 1)

c5        (0, 0, 0)

```

kontrolinis pavyzdys 9

būsena, reikšmingos linijos

```

cmp_prec      x
cmp01         (X, X)
cmp12         (X, X)
cmp23         (X, X)
cmp_exp12     (X, X)
insig1        ((X, X), (X, X), (X, X))
insig2        ((X, X), (X, X), (X, X))
insig3        ((X, X), (X, 0), (X, X))
sign1         (X, X, X, X)
sign2         (X, X, X, X)
sign3         (0, X, X, 1)
gr_i1         (X, X, X)
c_ei1         x
gr_d1         (X, X, X)
c_ed1         x
co_limit1     (X, X)
gr_i2         (X, X, X)
c_ei2         x
gr_d2         (X, X, X)
c_ed2         x
co_limit2     (X, X)
gr_a1         (X, X, X)
gr_a2         (X, X, X)
gr_a3         (X, X, X)

ins           [1; 1; 0; 0; 0; 0; 0; 0; 1]

```

mikroinstrukcijų linijos

```

c1
c1_1          (0, 0, 0, 1, 0, 0)
c1_2          [(0, 1); (0, 1); (1, 0); (0, 1)]
c1_3          [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c1_4          [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c1_5          [(0, 1); (1, 0); (0, 1)]
c1_6          [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c1_7          (0, [(0, 1); (1, 0)], [0; 0])
c1_8          (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c1_10         [0; 0; 0; 0; 1]

c2
c2_1          (0, 0, 0, 0, 1, 0)
c2_2          [(0, 1); (0, 1); (1, 0); (0, 1)]
c2_3          [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c2_4          [(0, 1); (1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_5          [(0, 1); (1, 0); (0, 1)]
c2_6          [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_7          (0, [(0, 1); (1, 0)], [0; 0])
c2_8          (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c2_10         [0; 0; 0; 0; 1]

c3
c3_1          (1, 0)
c3_2          [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
c3_3          (0, [(0, 1); (1, 0)], [1; 1])
c3_4          [0; 0; 0; 0; 1]
c3_5          [(1, 0); (0, 1)], (0)

c4            (0, 1)

c5            (0, 0, 0)

```

kontrolinis pavyzdys 10

būsena, reikšmingos linijos- sąlyga 1

cmp_prec	X
cmp01	(X, X)
cmp12	(X, X)
cmp23	(X, X)
cmp_exp12	(X, X)
insig1	((X, X), (X, X), (X, X))
insig2	((X, X), (X, X), (X, X))
insig3	((X, X), (X, 0), (X, X))
sign1	(X, X, X, X)
sign2	(0, X, X, X)
sign3	(0, X, X, 0)
gr_i1	(X, X, X)
c_ei1	X
gr_d1	(X, X, X)
c_ed1	X
co_limit1	(X, X)
gr_i2	(X, X, X)
c_ei2	X
gr_d2	(X, X, X)
c_ed2	X
co_limit2	(X, X)
gr_a1	(X, X, X)
gr_a2	(X, 0, X)
gr_a3	(X, X, X)

ins [1; 1; 0; 0; 0; 0; 0; 0; 1]

būsena, reikšmingos linijos - sąlyga 2

cmp_prec	X
cmp01	(X, X)
cmp12	(X, X)
cmp23	(X, X)
cmp_exp12	(X, X)
insig1	((X, X), (X, X), (X, X))
insig2	((X, X), (X, X), (X, X))
insig3	((X, X), (X, 0), (X, X))
sign1	(X, X, X, X)
sign2	(1, X, X, X)
sign3	(0, X, X, 0)
gr_i1	(X, X, X)
c_ei1	X
gr_d1	(X, X, X)
c_ed1	X
co_limit1	(X, X)
gr_i2	(X, X, X)
c_ei2	X
gr_d2	(X, X, X)
c_ed2	X
co_limit2	(X, X)
gr_a1	(X, X, X)
gr_a2	(X, 1, X)
gr_a3	(X, X, X)

ins [1; 1; 0; 0; 0; 0; 0; 0; 1]

mikroinstrukcijų linijos.

cl	
cl_1	(0, 0, 0, 0, 1, 0)
cl_2	((0, 1); (0, 1); (1, 0); (0, 1))

```

c1_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c1_4      [(0, 1); (0, 1); (1, 0); (0, 1); (0, 1); (0, 1)]
c1_5      [(0, 1); (1, 0); (0, 1)]
c1_6      [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1)]
c1_7      (0, [(0, 1); (1, 0)], [0; 0])
c1_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c1_10     [0; 0; 0; 0; 1]

c2
c2_1      (0, 0, 0, 0, 1, 0)
c2_2      [(0, 1); (0, 1); (1, 0); (0, 1)]
c2_3      [(0, 1); (0, 1); (0, 1); (0, 1); (1, 0)]
c2_4      [(0, 1); (1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_5      [(0, 1); (1, 0); (0, 1)]
c2_6      [(1, 0); (0, 1); (0, 1); (0, 1); (0, 1)]
c2_7      (0, [(0, 1); (1, 0)], [0; 0])
c2_8      (0, [(0, 1); (1, 0)], [0; 0; 0; 0])
c2_10     [0; 0; 0; 0; 1]

c3
c3_1      (1, 0)
c3_2      [(0, 1); (0, 1); (0, 1); (1, 0); (0, 1); (0, 1)]
c3_3      (0, [(0, 1); (1, 0)], [1; 1])
c3_4      [0; 0; 0; 0; 1]
c3_5      ([1, 0]; (0, 1)], 0)

c4      (0, 1)

c5      (0, 0, 0)

```

APIBRĖŽTIS

1. Būdas, aritmetinėms, loginėms ir panašioms operacijoms su skaitmeninio dydžio elementais atlikti, apibrėžiamas kaip įjungiantis šiuos veiksmus:

a) paduodantis įėjimo sąrašą, turintį iš išanksto apibrėžtą didžiausią dvejetainių žodžių skaičių, lygų didžiausiam elementų skaičiui įėjimo sąrašė, ir susidedantį iš skaitmeninių dydžių, kurie bus apdorojami, elementų, atstovaujamų užduoto ilgio dvejetainiu žodžiu, ir komandos informacijos, mažiausiai dalis kurios atstovaujama dvejetainiu žodžiu, parodančiu atliekamos operacijos tipą,

b) perduodantis, mažiausiai, dvejetainius žodžius, atstovaujančius sąrašo skaitmeninio dydžio elementus, pateiktus apdorojimui, per valdomų pertvarkančių ir sujungiančių grandinių tinklą,

c) atliekantis operacijas su įėjimo sąrašo skaitmeninio dydžio elementais tiesiogiai valdant, naudojantis komandų informacija, minėtas grandines, tokiu būdu, kad operacijos atliekamos perrašant ir/arba sujungiant apdorojamus dvejetainius žodžius,

d) atspindintis rezultatą kaip išėjimo sąrašą, turintį su įėjimo sąrašu vienodą formatą,

e) jeigu reikalinga, pakartojantis rezultato išėjimo sąrašė perdavimą į įėjimo sąrašą iki gaunamas tinkamas rezultatas.

2. Būdas, pagal apibrėžties punktą 1, apibrėžiamas taip, kad kiekvienu perrašymu, pakartojamo perrašymo operacijoje, sąrašas perrašomas įjungiant pakeistą komandos kodą, einantį prieš dydžio žodį arba žodžius, jeigu tai reikalinga atliekamiems skaičiavimams.

3. Būdas, pagal apibrėžties punktą 1, apibrėžiamas taip, kad, perrašant tiesiai į rezultatą, dydis į išėjimo sąrašą pateikiamas pakeistas minėto komandos kodo.

4. Būdas, pagal kiekvieną prieš einantį apibrėžties punktą, apibrėžiamas taip, kad didžiausias elementų įėjimo sąrašė ir išėjimo sąrašė skaičius yra keturi.

5. Būdas, pagal kiekvieną prieš einantį apibrėžties punktą, apibrėžiamas taip, kad skaitmeninio dydžio žodžiai pateikiami koduotoje formoje prieš vykdomą apdorojimą.

6. Būdas, pagal kiekvieną prieš einantį apibrėžties punktą, apibrėžiamas taip, kad žodžiai, atspindintys skaitmeninio dydžio žodžius, pateikiami, kodavimo būdu sutankintu, skaitmeninio dydžiu, t.y. kiekvienam koduoto dydžio atspindėjimui atitinka tik vienas skaitmeninis dydis.

7. Būdas, pagal kiekvieną prieš einantį apibrėžties punktą, apibrėžiamas taip, kad binarinis žodis, atspindintis sveiką dydį, gaunamas pirmojo tipo kodavimu.

8. Būdas, pagal kiekvieną prieš einantį apibrėžties punktą, apibrėžiamas taip, kad binarinis žodis, atspindintis slankaus kabelio dydį, gaunamas antrojo tipo kodavimu.

9. Būdas, pagal kiekvieną prieš einantį apibrėžties punktą, apibrėžiamas taip, kad binarinis žodis, atspindintis sveiką dydį, gaunamas pirmojo tipo kodavimu ir binarinis žodis, atspindintis slankaus kabelio dydį, gaunamas antrojo tipo kodavimu, ir, kad minėti kodavimai yra tokie, kad po kodavimo slankaus kabelio skaitmeninio dydžio atspindėjimai didėja ta pačia tvarka kaip ir sveiko skaitmeninio dydžio atspindėjimai.

10. Būdas, pagal kiekvieną prieš einantį apibrėžties punktą, apibrėžiamas taip, kad binarinis koduoto slankaus kabelio dydžio atspindėjimas įjungia kodo lauką, eksponentės lauką ir mantisės lauką, taip, kad kodo lauke pateikiama padalinimo tarp kintamo ilgio eksponentės ir mantisės laukų padėtis.

11. Būdas, pagal kiekvieną nuo 6 iki 10 apibrėžties punktą, apibrėžiamas taip, kad minėtas sutankintas atspindėjimas gaunamas įjungian virtualų „1“, t.y. fiziškai neatspindimą „1“, prieš slankaus kabelio dydį atspindinčio žodžio mantisės lauko dalį.

12. Būdas, pagal kiekvieną nuo 6 iki 11 apibrėžties punktą, apibrėžiamas taip, kad minėti dydžiai koduojami sekančiu būdu: a) dydis atspindimas bitų seka, kurioje kiekvienas bitas yra „1“ arba „0“, užkoduota pradedant nuo skaitmeninės nulio reikšmės, b) kiekvienas galimas dydis minėtame atspindėjime atitinka tik vieną slankaus kabelio dydį c) slankaus kabelio dydis, kurio atspindėta reikšmė didesnė nei kito dydžio atspindėta reikšmė, yra didesnis už kitą, d) minėtas slankaus kabelio dydžio atspindėjimas įjungia ženklo bitą visam slankaus kabelio dydžiui, slankaus kabelio dydžio eksponentės ženklo bitą, kodą apsprendžiantį eksponentės ilgį, pačią eksponentę, turinčią kintamą ilgį, ir mantisę, turinčią kintamą ilgį, esant pastoviam eksponentės ir mantisės bendram ilgiui.

13. Būdas, pagal apibrėžties punktą 12, apibrėžiamas taip, kad „1“-to atspindėjimas reikšmingiausiam bite ir „0“ kituose bituose atspindi slankaus kabelio dydį lygų „0“.

14. Būdas, pagal apibrėžties punktą 13, apibrėžiamas taip, kad visi atspindėjimai, nesutampantys su slankaus kabelio dydžiu „0“, atitinka dydį 2^{e+1} * mantisė, čia e yra eksponentė, kuri minėtame slankaus kabelio dydžių atspindėjime, teigiamiems slankaus kabelio dydžiams didesnė arba lygi 1, neigiamiems slankaus kabelio dydžiams mažesnė už -1, ir ten kur kinta tarp 1,0 ir -1,0 yra atspindima invertuota, išskyrus slankaus kabelio dydį „0“, minėta mantisė kinta tarp -1 ir -2 neigiamų dydžių atspindėjimas, t.y. neigiamiems skaičiams, ir tarp +1 ir +2 teigiamo ženklo atspindėjimams, t.y. teigiamiems skaičiams.

15. Būdas, pagal kiekvieną prieš einantį apibrėžties punktą, apibrėžiamas taip, kad komanda vykdoma atliekant lygiagrečius, eilės galimų kombinacijų su dydžiais, naudojamais kaip tarpinės galutinio dydžio stadijos, skaičiavimus,

pasirenkant tarpinę stadiją tolimesniam apdorojimui, jeigu tai reikalinga, galutiniam rezultatui gauti.

16. Būdas, pagal kiekvieną apibrėžties punktą nuo 6 iki 15 apibrėžiamas taip, kad pirmojo slankaus kabelio dydžio A ir antrojo slankaus kabelio dydžio B sudėties/atimties operacijos atliekamos sekančiai: išlyginant dydį A su dydžiu B, kai A dydžio eksponentė yra mažesnė už B dydžio eksponentę, išlyginant dydį B su dydžiu A, kai B dydžio eksponentė yra mažesnė už A dydžio eksponentę, išlyginant minėtus dydžius sudėties/atimties operacijoje, normalizuojant sudėties/atimties rezultata. 17. Būdas pagal apibrėžties punktą 16 apibrėžiamas kaip, kad sudėties/atimties operacijos skaičiuojamas lygiagrečiai kiekvienoje sudėties/atimties operacijos stadijoje ir pasirenkama, tarp lygiagrečiai atliekamų operacijų, tinkama skaičiuojamai operacijos stadijai.

18. Būdas, pagal kiekvieną apibrėžties punktą nuo 12 iki 17 apibrėžiamas taip, kad atliekami sekantys veiksmai vykdant pirmojo slankaus kabelio dydžio A ir antrojo slankaus kabelio dydžio B sudėties/atimties operaciją:

išlyginamas dydis A su dydžiu B ir padidinamas/sumažinamas jų mantisės tikslumas, kai A dydžio eksponentė yra mažesnė už B dydžio eksponentę,

išlyginamas dydis B su dydžiu A ir padidinamas/sumažinamas jų mantisės tikslumas, kai B dydžio eksponentė yra mažesnė už A dydžio eksponentę,

sudėties/atimties operacijoje minėti dydžiai išlyginami normalizuojant sudėties/atimties rezultata.

19. Būdas, pagal apibrėžties punktą 18 apibrėžiamas taip, kad sudėties/atimties operacijos skaičiuojamas lygiagrečiai kiekvienoje sudėties/atimties operacijos stadijoje ir pasirenkama, tarp lygiagrečiai atliekamų operacijų, tinkama skaičiuojamai operacijos stadijai.

20. Būdas, pagal kiekvieną apibrėžties punktą nuo 6 iki 19 apibrėžiamas taip, kad dviejų slankaus kabelio dydžių B ir C daugyba atliekama perrašant sandaugą $0,0+B \cdot C=V$, kur V yra pastovus dydis, perrašomas keleto stadijų, turinčių tarpinį dydį $A+B \cdot C=V$, kur A prieš buvęs V dydis, o eksponentė C i yra pritaikyta sudėčiai/atimčiai, metu, taip, kad dydžio B i eksponentė galutinai taptų „0“, o išėjimo galutinio rezultato eksponentė būtų galutinė C i eksponentė.

21. Skaitmeninis aritmetinis įtaisas aritmetinėms, loginėms ir panašioms operacijoms su skaitmeninio dydžio elementais įjungia:

a) įėjimą, sudarytą iš eilės magistralių ($v0i, v1i, v2i, v3i$), kiekviena kuri veikdama yra sujungta įėjimo sąrašo elementu, sąrašas įjungia skaitmeninio dydžio elementus ($v1i, v2i, v3i$) apdorojimui ir komandas, parodančias kokio tipo operacijos turi būti atliktos, kiekvienas skaitmeninio dydžio elementas, atspindintis skaitmeninį, žodį yra iš anksto užduoto ilgio, mažiausiai dalis komandų informacijos atspindima kaip dvejetainis žodis ($v0i$) didžiausias dvejetainių žodžių skaičius yra lygus didžiausiam sąrašo elementų skaičiui įėjimo sąraše,

b) apdorojimo įtaisas įjungia tinklą valdomų pertvarkančių ir sujungiančių grandinių, prie kurių prijungtos minėtos magistralės, ir pritaikytų vykdyti komandų informacijoje, gaunamoje iš įėjimo sąrašo, nurodomas operacijas, atlikdamas jas perrašant ir/arba sujungiant apdorojamus dvejetainius žodžius, ir pateikiant perrašytą rezultata į išėjimą, sudarytą iš eilės išėjimo magistralių, struktūra ir skaičiumi sutampančių su įėjimo magistralėmis,

c) galutinio rezultato tikrinimo priemones ($12A - 12N, 13A - 13N, 10, 21, 22, 221, 222, 223, 27, A_{STATE}$) apdorojimo įtaise, tikrinančias ar galutinis rezultatas gautas ir atliekančias perdavimą iš išėjimo magistralių į įėjimo magistralės, pakartojant sąrašo perrašymą į apdorojimo įtaisa iki galutinis rezultatas bus pasiektas.

22. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktą 21, apibrėžiamas taip, kad eilė grandinių ($21, 22, 221, 23, 24, 25, 28, 29, 30, 35, 36, 37$), kiekviena pritaikyta atlikti jai būdingas operacijas su skaitmeninių dydžių atspindėjimais iš įėjimo magistralių, išvedant operacijų rezultatus lygiagrečiai, ir, kad valdymo įtaisas (27), gaudamas komandų informaciją išsirenka iš visų gautų rezultatų tinkamus esančios komandos vykdymui.

23. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktą 21, apibrėžiamas taip, kad sąrašas įjungia panaudojimo funkciją, kurioje vienas iš elementų yra komandos kodas, o likusieji komandos argumentai, apdorojimo įtaisas atlieka komandą perrašydamas ir pakartotinai perrašydamas instrukcijos kodą iš išėjimo į įėjimą iki pasiekiamas galutinis rezultatas ir kad kiekvieno perrašymo metu apdorojimo įtaisas yra nustatomas perrašyti sąrašą su pakeistu komandos kodu žodžiu, einančiu prieš dydžių žodžius, jeigu tai reikalinga atliekamiems skaičiavimams.

24. Skaitmeninis aritmetinis, įtaisas pagal apibrėžties punktą 24 arba 25, apibrėžiamas taip, kad skaitmeninių dydžių žodžiai įtaise prieš apdorojimą atspindimi koduotoje formoje ir, kad į valdymo įtaisa (27) gauna informaciją, nurodančią atspindėjimo kodavimo tipą.

25. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktą 24, apibrėžiamas taip, kad apdorojimo įtaisa, pritaikytas naudoti kodus, atspindinčius skaitmeninius dydžius į sutankintą atspindėjimą, t.y. kiekvieną koduotą dydžio atspindėjimą atitinka tiksliai vienas interpretuojamas dydis.

26. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktą 24 arba 25, apibrėžiamas taip, kad apdorojimo įtaisa, pritaikytas naudoti pirmojo tipo binarinio žodžio kodavimą atspindint sveikus dydžius,

27. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktus nuo 24 iki 26, apibrėžiamas taip, kad apdorojimo įtaisa, pritaikytas naudoti antrojo tipo binarinio žodžio kodavimą atspindint slankaus kabelio dydžius,

28. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktus nuo 24 iki 27, apibrėžiamas taip, kad apdorojimo įtaisa, pritaikytas naudoti pirmojo tipo binarinio žodžio kodavimą atspindint sveikus dydžius ir antrojo tipo binarinio žodžio kodavimą atspindint slankaus kabelio dydžius ir, kad kodavimas yra toks, kad slankaus kabelio dydis atspindimas tokia

pačia tvarka kaip ir sveikas dydis,

29. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktą 27 arba 28, apibrėžiamas taip, kad binarinis koduoto slankaus kablelio dydžio atspindėjimas įjungia ženklą, eksponentės ženklą ir kodo lauką, eksponentės lauką ir mantisės lauką, taip, kad eksponentės ženklas ir kodo laukas parodo padalinimo padėtį tarp eksponentės ir mantisės kintamo ilgio laukų.

30. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktą 28 arba 29, apibrėžiamas taip, kad minėti žodžiai atspindintys skaitmeninių dydžių žodžius yra gauti koduojant skaitmeninius dydžius į sutankintą atspindėjimą, t.y. kiekvieną koduotą dydžio atspindėjimą atitinka tiksliai vienas interpretuojamas dydis.

31. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktą 30, apibrėžiamas taip, kad minėtas sutankintas atspindėjimas gaunamas įjungiant virtualų „1“, t.y. fiziškai neatspindimą „1“, prieš slankaus kablelio dydį atspindinčio žodžio mantisės lauko dalį.

32. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktą 29 arba 30, apibrėžiamas taip, kad minėtame atspindėjime slankaus kablelio dydis yra suskirstomas į bitų grupes, vadinamas tikslumo grupes, kuri kiekviena užima iš anksto užduotą bitų skaičių, slankaus kablelio dydžio padalinimas tarp eksponentės dalies ir mantisės dalies, atliekamas tarp dviejų tikslumo grupių, minimame atspindėjime.

33. Skaitmeninis aritmetinis įtaisas, pagal kiekvieną iš apibrėžties punktų nuo 21 iki 32, apibrėžiamas taip, kad į mažiausiai vieno operando įtaiso (r_1 , r_2 , r_3) įėjimus paduodami mažiausiai du skaitmeninio dydžio žodžiai ir skaitmeninio dydžio žodis, iš tokių operacijas vykdančių grandinių (35, 36, 37), kurių išėjimai pritaikyti tolimesniam įtaiso galutinio išėjimo dydžio formavimui, ir valdomo valdymo grandinių tolimesniam duomenų operando įtaiso įėjimuose, perrašant skaitmeninio dydžio žodius, apdorojimui.

34. Skaitmeninis aritmetinis įtaisas, pagal kiekvieną iš apibrėžties punktų nuo 21 iki 32, apibrėžiamas taip, kad išėjimo signalai iš operacijas atliekančių grandinių paduodami į valdymo įtaiso grandinių (27) įėjimus, ir yra apdorojami valdymo įtaiso grandinėse kartu su komandų informacija formuoti dvejetainius signalus mažiausiai vieno operando valdymui.

35. Skaitmeninis aritmetinis įtaisas, pagal kiekvieną iš apibrėžties punktų nuo 33 iki 34, apibrėžiamas taip, kad valdymo įtaiso (27) grandinės formuoja signalus mažiausiai pirmajame operacijos fazės intervale ir išėjimo fazės intervale, vieno duomenų vidinio perdavimo iš įėjimo į išėjimą laikotarpiu, ir, kad galimi perkėlimo signalai iš elementų operando grandinėje, valdomų pirmosios operacijos fazės intervalo metu, yra paduodami į valdymo įtaiso grandines (27), kurios valdo, galimų perkėlimo signalų prieš išėjimo fazės intervalą metu, vidinius operando grandinės elementus, turinčius praėjimus į išėjimą.

36. Skaitmeninis aritmetinis įtaisas, pagal kiekvieną iš apibrėžties punktų nuo 21 iki 38, apibrėžiamas taip, kad valdymo įtaiso (27) grandinėmis yra loginių elementų matrica.

37. Skaitmeninis aritmetinis įtaisas, pagal kiekvieną iš apibrėžties punktų nuo 21 iki 38, apibrėžiamas taip, kad didžiausias įėjimo ir išėjimo sąrašo ilgis yra keturi elementai, kiekvienas paduodamas į magistrales.

38. Skaitmeninis aritmetinis įtaisas, pagal kiekvieną iš apibrėžties punktų nuo 21 iki 37, apibrėžiamas taip, kad skaitmeninių dydžių atspindėjimai yra suskirstyti į bitų grupes, vadinamas tikslumo grupes, kuri kiekviena užima iš anksto užduotą bitų skaičių, slankaus kablelio dydžio padalinimas tarp eksponentės dalies ir mantisės dalies, atliekamas tarp dviejų tikslumo grupių, minimame atspindėjime; ir, kad mažiausiai vienas tikslumo dekoderis yra pritaikytas gauti mažiausiai dalį į jo įėjimus paduodamos informacijos apie eksponentės/mantisės padalinimą, paduodant, atitinkančius padalinimą, valdymo išėjimo signalus į elementus, atliekančius operacijas su slankaus kablelio dydžiais.

39. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktą 38, apibrėžiamas taip, kad yra du tikslumo dekoderiai.

40. Skaitmeninis aritmetinis įtaisas, pagal apibrėžties punktą 38 arba 39, apibrėžiamas taip, kad trisdešimt dviejų bitų atspindėjime naudojamos aštuonios tikslumo grupės.

41. Skaitmeninis aritmetinis įtaisas, pagal kiekvieną iš apibrėžties punktų 29 arba 40, apibrėžiamas taip, kad slankaus kablelio dydžio atspindėjimo informacija paduodama į kintamo ilgio kodo lauką.

Į KITUS
ĮTAISUS

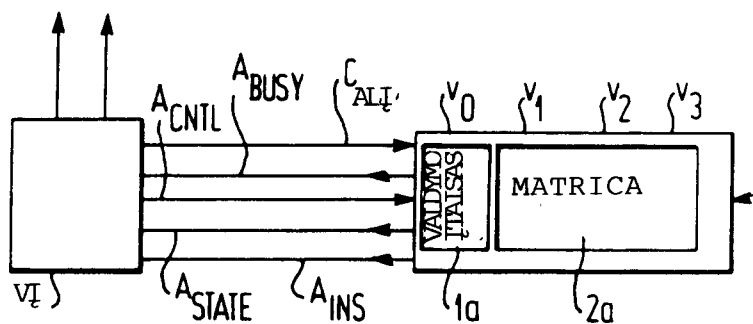


FIG. 1A

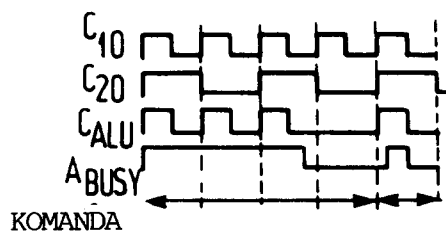


FIG.1B

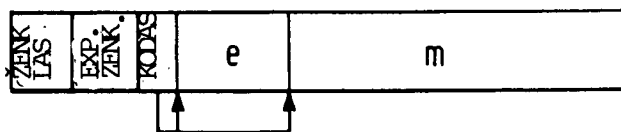
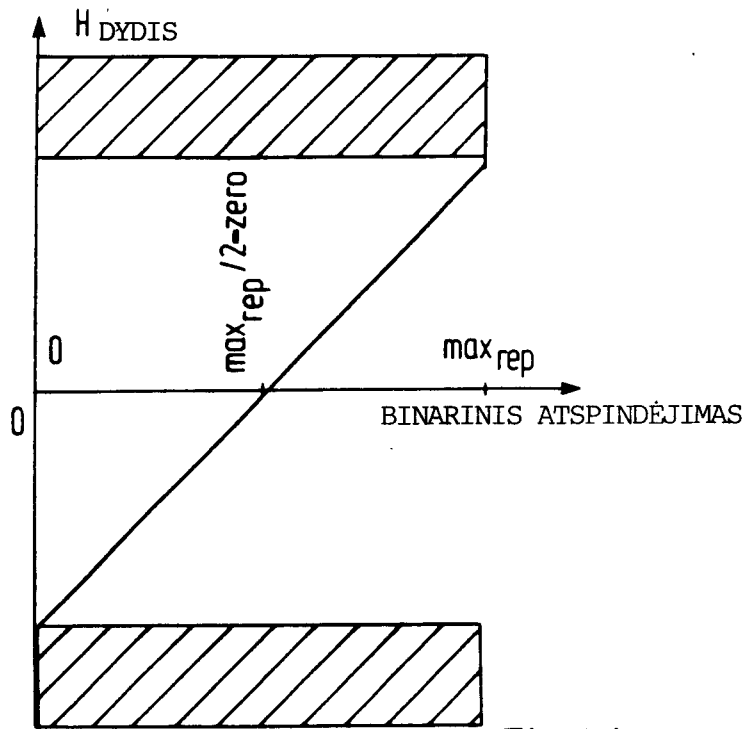


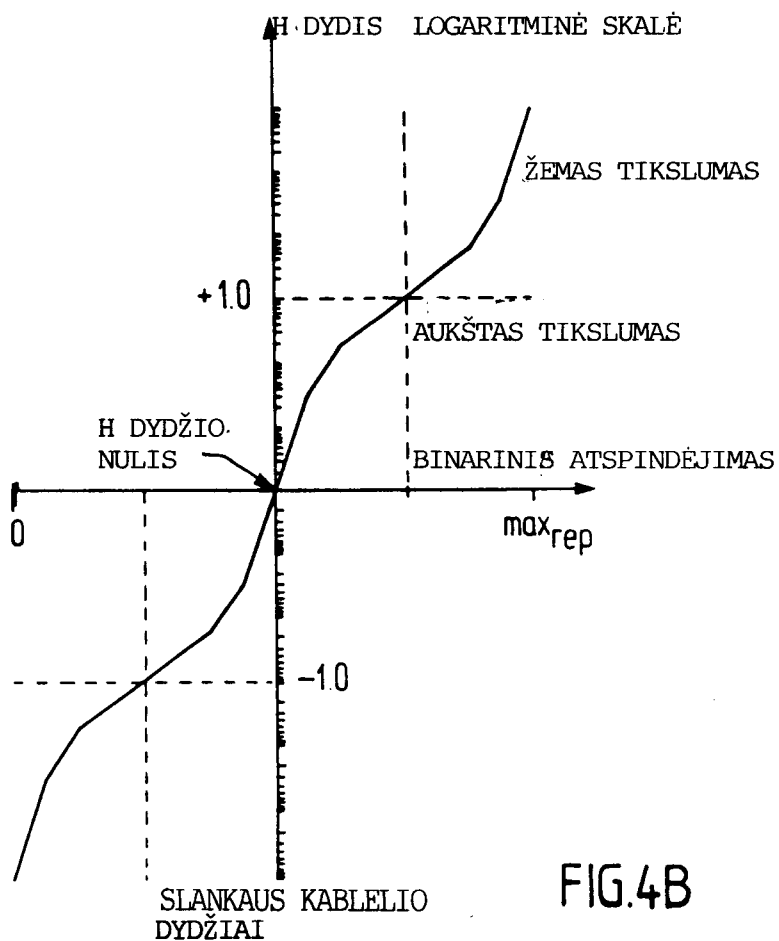
FIG. 3



FIG. 2



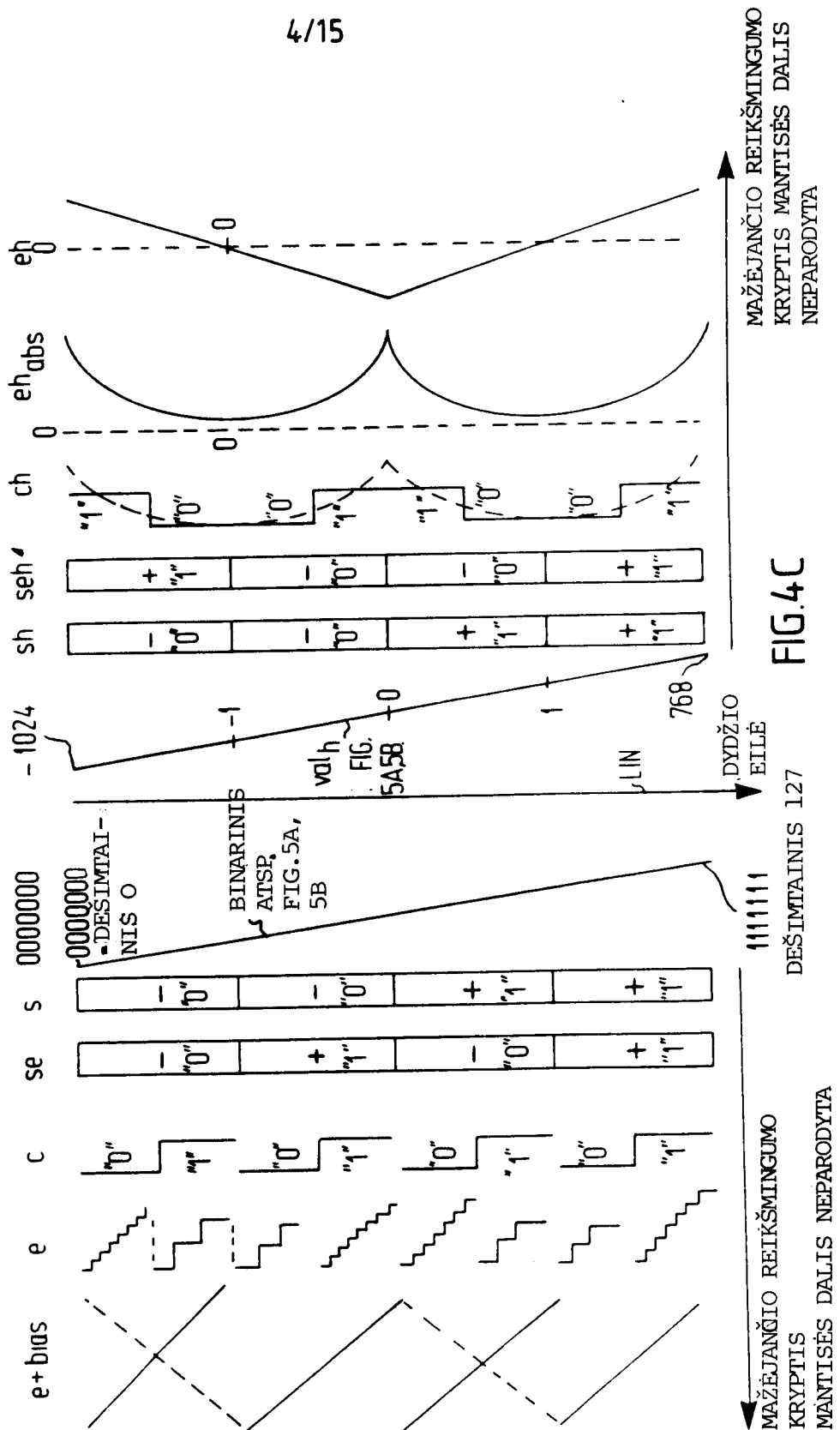
SVEIKI DYDŽIAI



SLANKAUS KABELIO DYDŽIO ATSPINDĖJIMAS

BINARINIS ATSPINDĖJIMAS

H DYDIS



IPRASTAS		DEŠIMTAINIS BINARINIO DYDŽIO											
BINARINIS	ATSPIN	ATSPIND										m _h	val _h
		0	1	2	3	4	5	6	7	8	9		
0000000	0	-	-	-	-	-	-	-	-	-	-	-2	-1024
0000001	1	-	-	-	-	-	-	-	-	-	-	-1.5	-768
0000010	2	-	-	-	-	-	-	-	-	-	-	-2	-512
0000011	3	-	-	-	-	-	-	-	-	-	-	-1.5	-384
0000100	4	-	-	-	-	-	-	-	-	-	-	-2	-256
0000101	5	-	-	-	-	-	-	-	-	-	-	-1.5	-192
0000110	6	-	-	-	-	-	-	-	-	-	-	-2	-128
0000111	7	-	-	-	-	-	-	-	-	-	-	-1.5	-96
0001000	8	-	-	-	-	-	-	-	-	-	-	-2	-64
0001001	9	-	-	-	-	-	-	-	-	-	-	-1.5	-48
0001010	10	-	-	-	-	-	-	-	-	-	-	-2	-32
0001011	11	-	-	-	-	-	-	-	-	-	-	-1.5	-24
0001100	12	-	-	-	-	-	-	-	-	-	-	-2	-16
0001101	13	-	-	-	-	-	-	-	-	-	-	-1.5	-12
0001110	14	-	-	-	-	-	-	-	-	-	-	-2	-8
0001111	15	-	-	-	-	-	-	-	-	-	-	-1.5	-6
0010000	16	-	-	-	-	-	-	-	-	-	-	-2	-4
0010001	17	-	-	-	-	-	-	-	-	-	-	-1.875	-3.75
0010010	18	-	-	-	-	-	-	-	-	-	-	-1.75	-3.5
0010011	19	-	-	-	-	-	-	-	-	-	-	-1.625	-3.25
0010100	20	-	-	-	-	-	-	-	-	-	-	-1.5	-3
0010101	21	-	-	-	-	-	-	-	-	-	-	-1.375	-2.75
0010110	22	-	-	-	-	-	-	-	-	-	-	-1.25	-2.5
0010111	23	-	-	-	-	-	-	-	-	-	-	-1.125	-2.25
0011000	24	-	-	-	-	-	-	-	-	-	-	-2	-2
0011001	25	-	-	-	-	-	-	-	-	-	-	-1.875	-1.875
0011010	26	-	-	-	-	-	-	-	-	-	-	-1.75	-1.75
0011011	27	-	-	-	-	-	-	-	-	-	-	-1.625	-1.625
0011100	28	-	-	-	-	-	-	-	-	-	-	-1.5	-1.5
0011101	29	-	-	-	-	-	-	-	-	-	-	-1.375	-1.375
0011110	30	-	-	-	-	-	-	-	-	-	-	-1.25	-1.25
0011111	31	-	-	-	-	-	-	-	-	-	-	-1.125	-1.125

FIG.5A

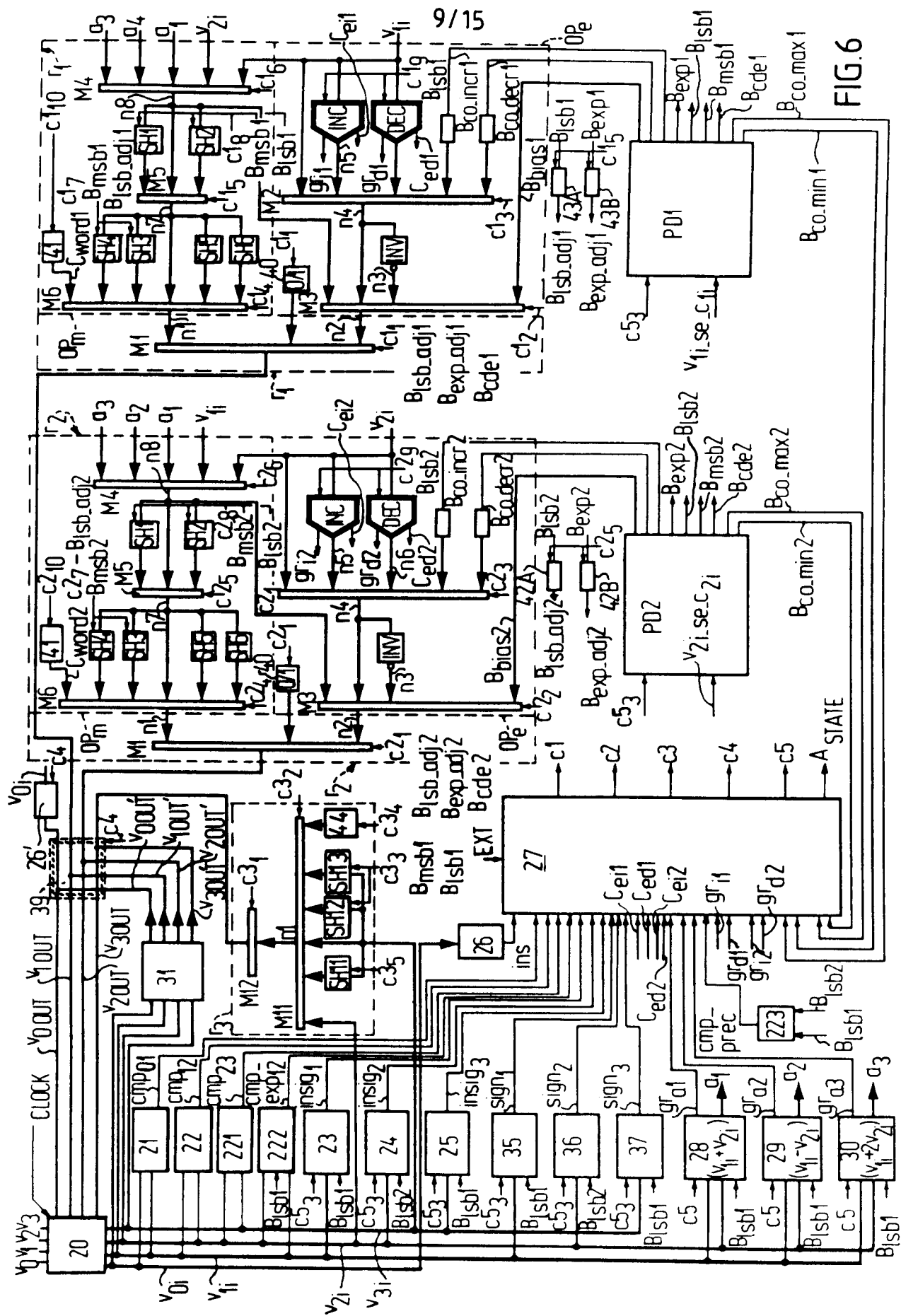
IPRASTAS BINARINIS ATSPINDĖJIMAS	DEŠIMTAINIS BINARINIO DYDŽIO ATSPIN.	s, sh, se, seh, c, ch, e, eh, e _h	m _d , m _h	m _h	val _h
0100000	32	->	0,000	-2	-1
0100001	33	->	0,000	-1.875	-.9375
0100010	34	->	0,000	-1.75	-.875
0100011	35	->	0,000	-1.625	-.8125
0100100	36	->	0,000	-1.5	-.75
0100101	37	->	0,000	-1.375	-.6875
0100110	38	->	0,000	-1.25	-.625
0100111	39	->	0,000	-1.125	-.5625
0101000	40	->	0,000	-2	-.5
0101001	41	->	0,000	-1.875	-.46875
0101010	42	->	0,000	-1.75	-.4375
0101011	43	->	0,000	-1.625	-.40625
0101100	44	->	0,000	-1.5	-.375
0101101	45	->	0,000	-1.375	-.34375
0101110	46	->	0,000	-1.25	-.3125
0101111	47	->	0,000	-1.125	-.28125
0110000	48	->	0,000	-2	-.25
0110001	49	->	0,000	-1.5	-.1875
0110010	50	->	0,000	-2	-.125
0110011	51	->	0,000	-1.5	-0.09375
0110100	52	->	0,000	-2	-0.0625
0110101	53	->	0,000	-1.5	-0.046875
0110110	54	->	0,000	-2	-0.03125
0110111	55	->	0,000	-1.5	-0.0234375
0111000	56	->	0,000	-2	-0.015625
0111001	57	->	0,000	-1.5	-0.0117188
0111010	58	->	0,000	-2	-0.0078125
0111011	59	->	0,000	-1.5	-0.00585938
0111100	60	->	0,000	-2	-0.00390625
0111101	61	->	0,000	-1.5	-0.00292969
0111110	62	->	0,000	-2	-0.00195312
0111111	63	->	0,000	-1.5	-0.00146484

FIG.5B

ATSPINĒJIMAS

2

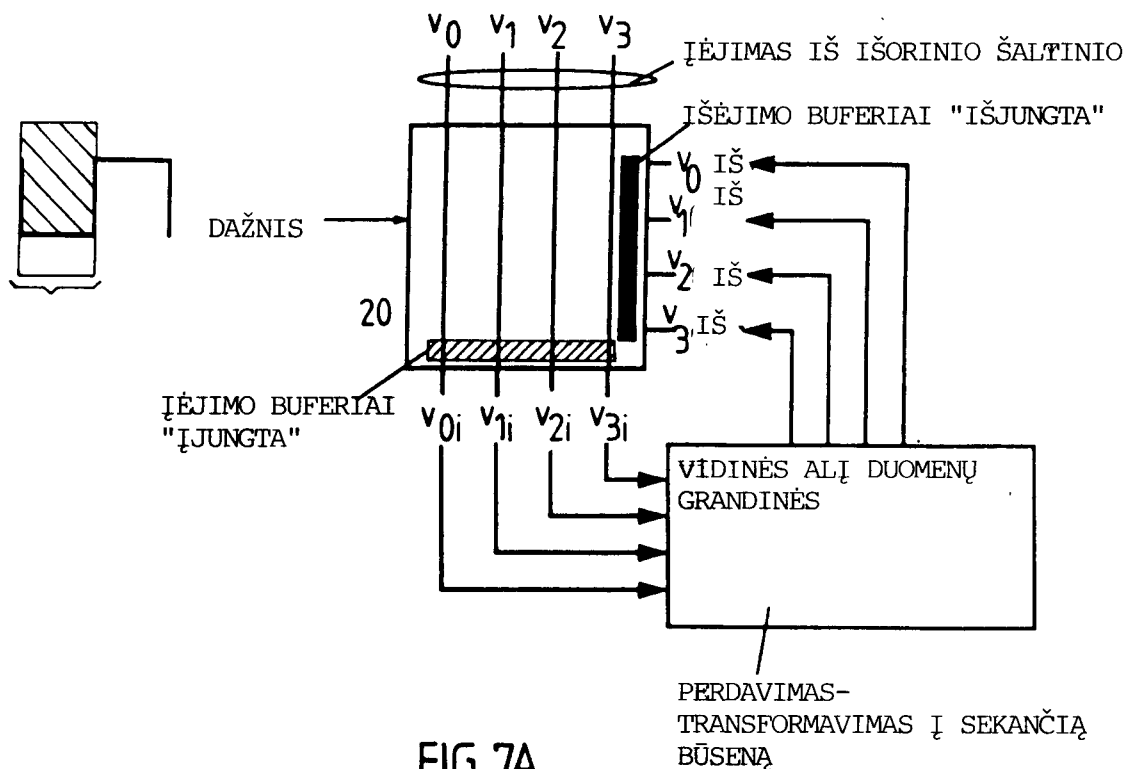
FIG. 5C



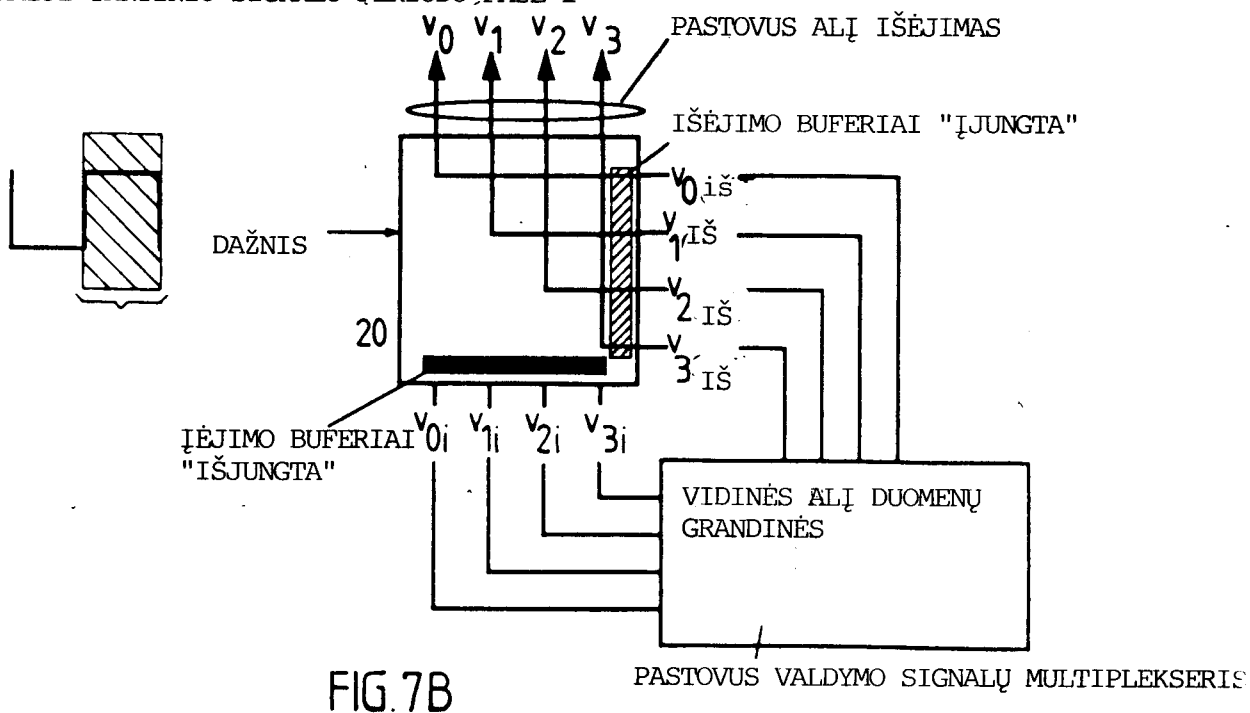
10/15

ĮĖJIMO/IŠĖJIMO BUFERIO 20 TAKTAVIMAS

PIRMOJI TAKTINIO SIGNALO PERIODO FAZĖ 0 KOMANDOMS

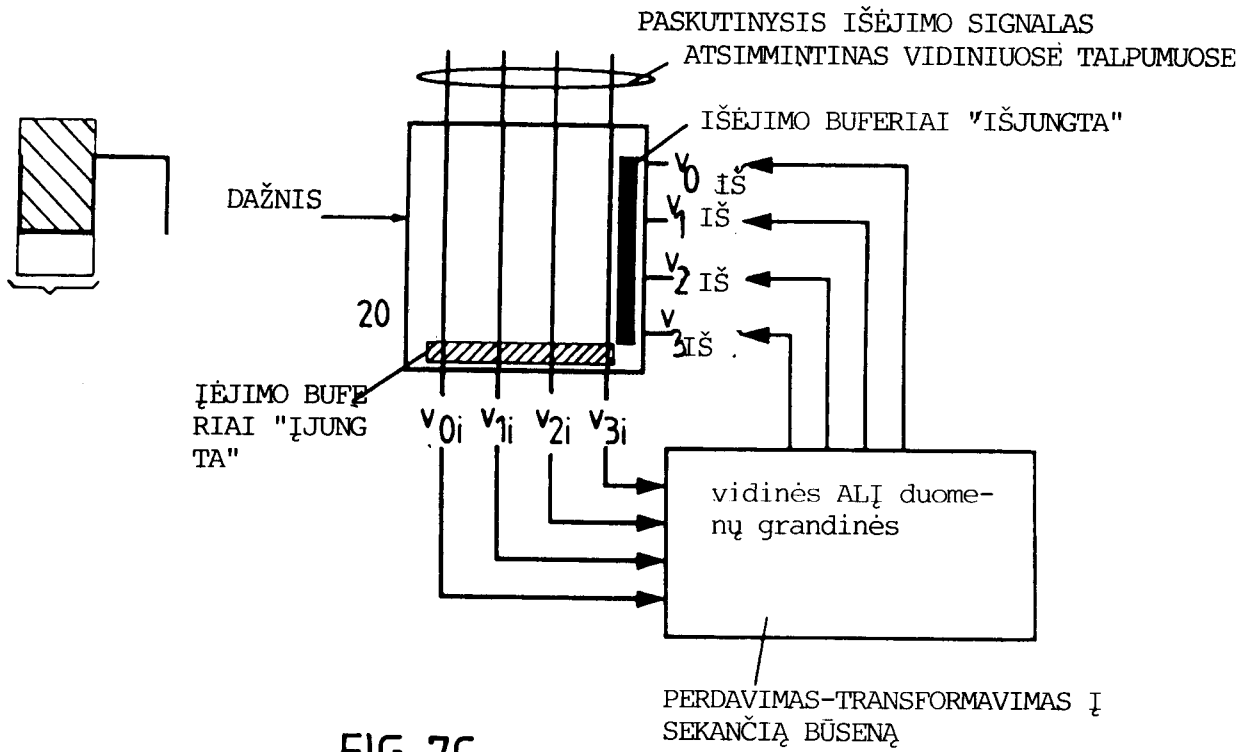


PIRMOJI TAKTINIO SIGNALO PERIODO FAZĖ 1



11/15

LIKUSIEJI TAKTINIO SIGNALO FAZĖS 0 PERIODAI DAUGIACIKLEI KOMANDAI



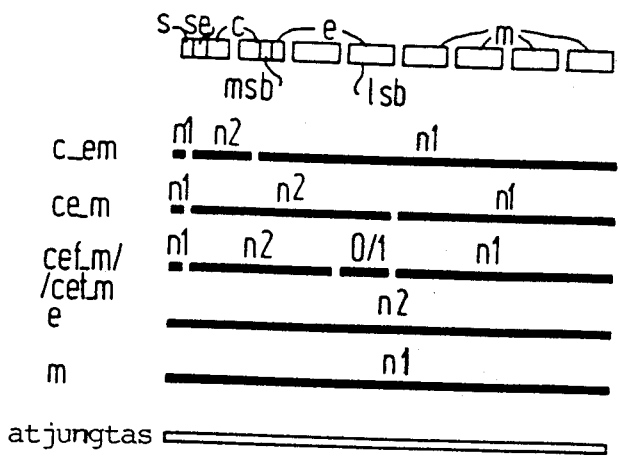


FIG.8

NORMALUS KODAS m 1.625
 0.5 0.25 0.125 0.0625
 KEIČIAMAS KODAS m 0.625
 0.5 0.25 0.125 0.0625
 REALIŲ DYDŽIŲ KODAVIMAS

FIG.10

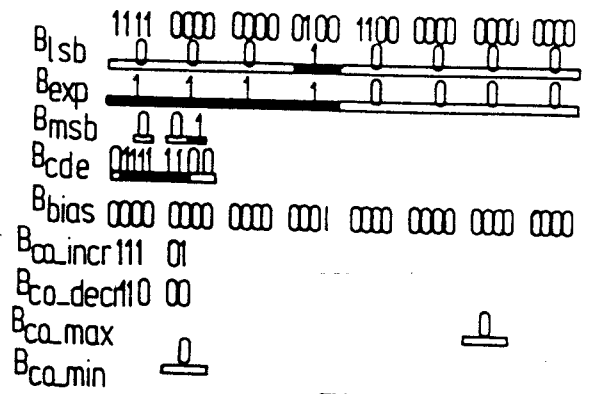


FIG.9

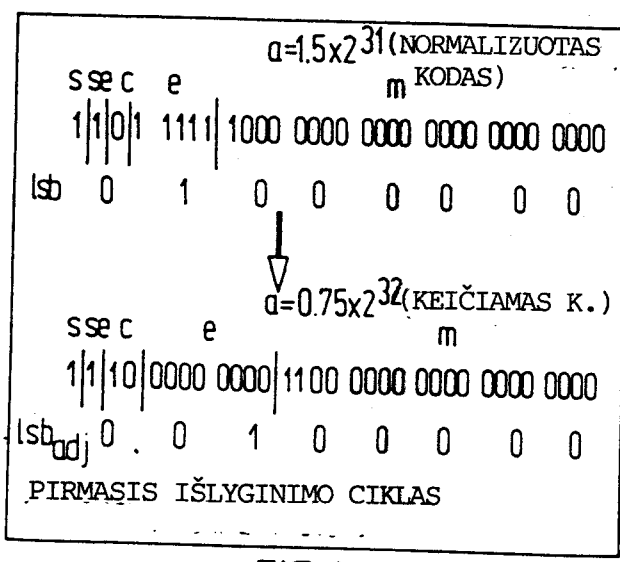


FIG.11

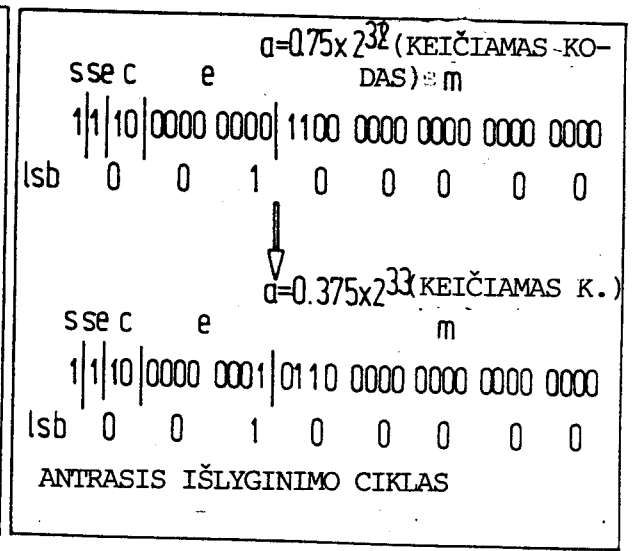


FIG.12

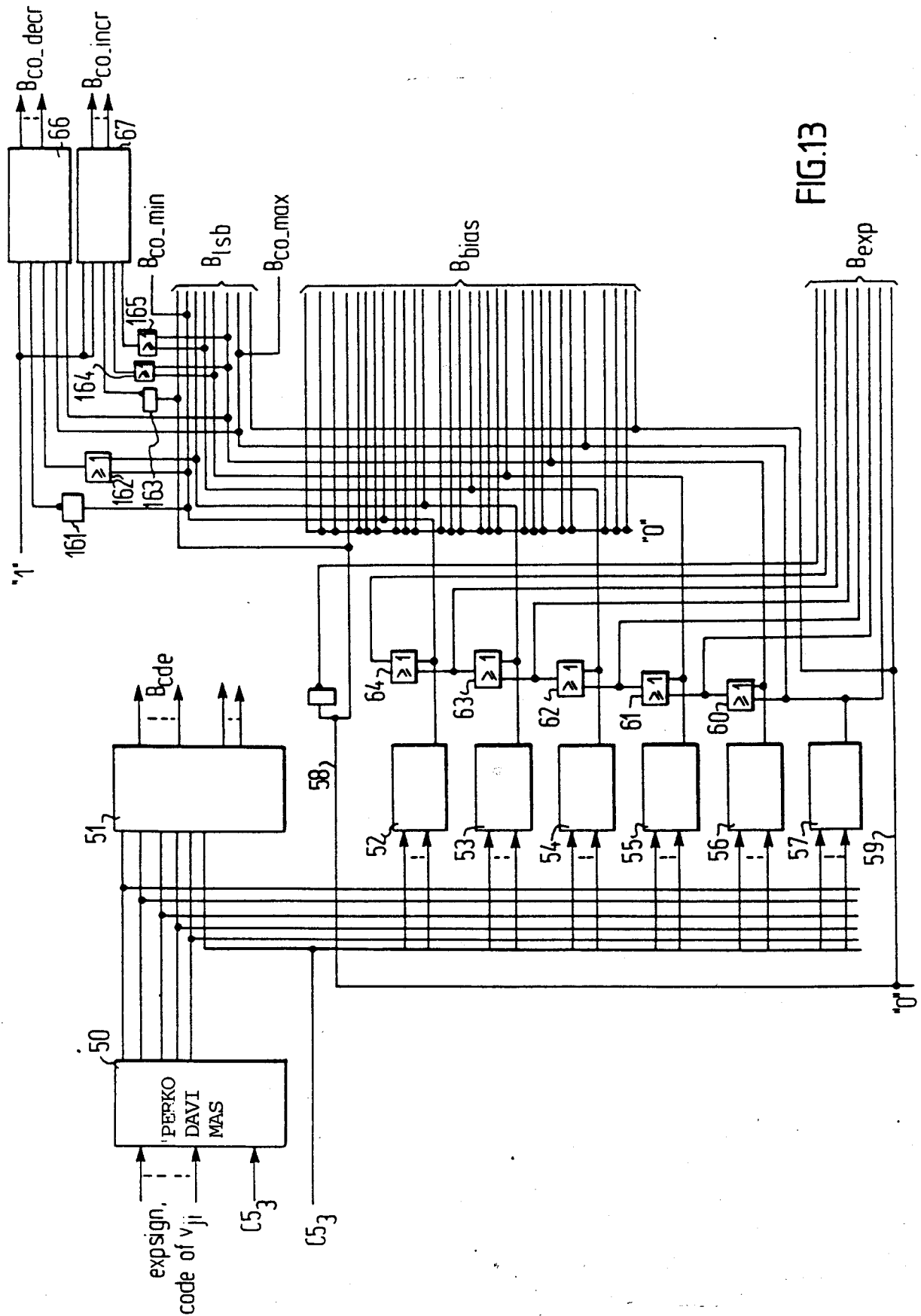
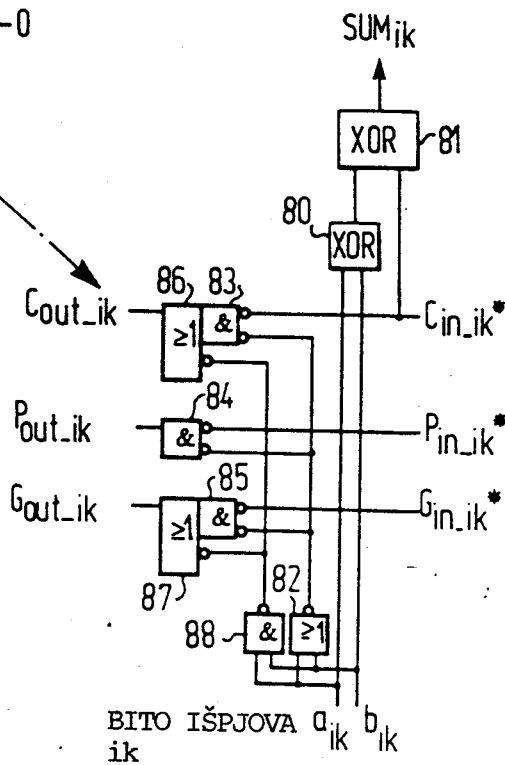
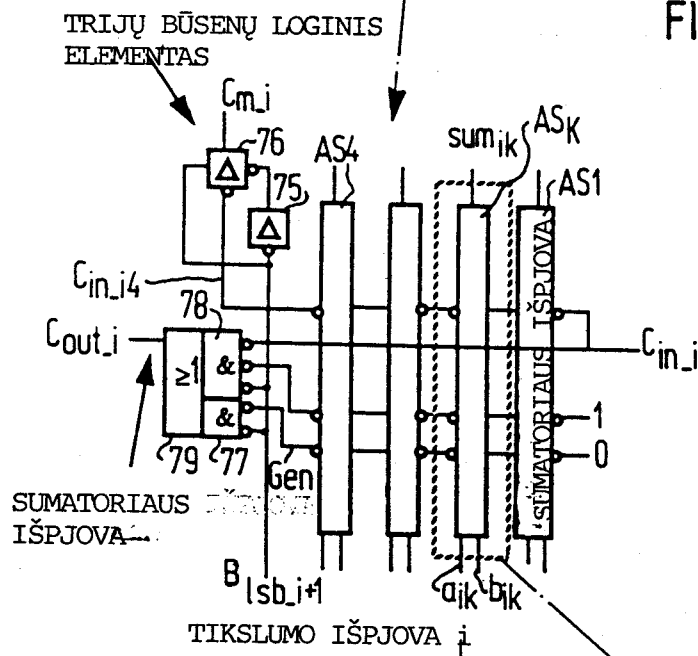
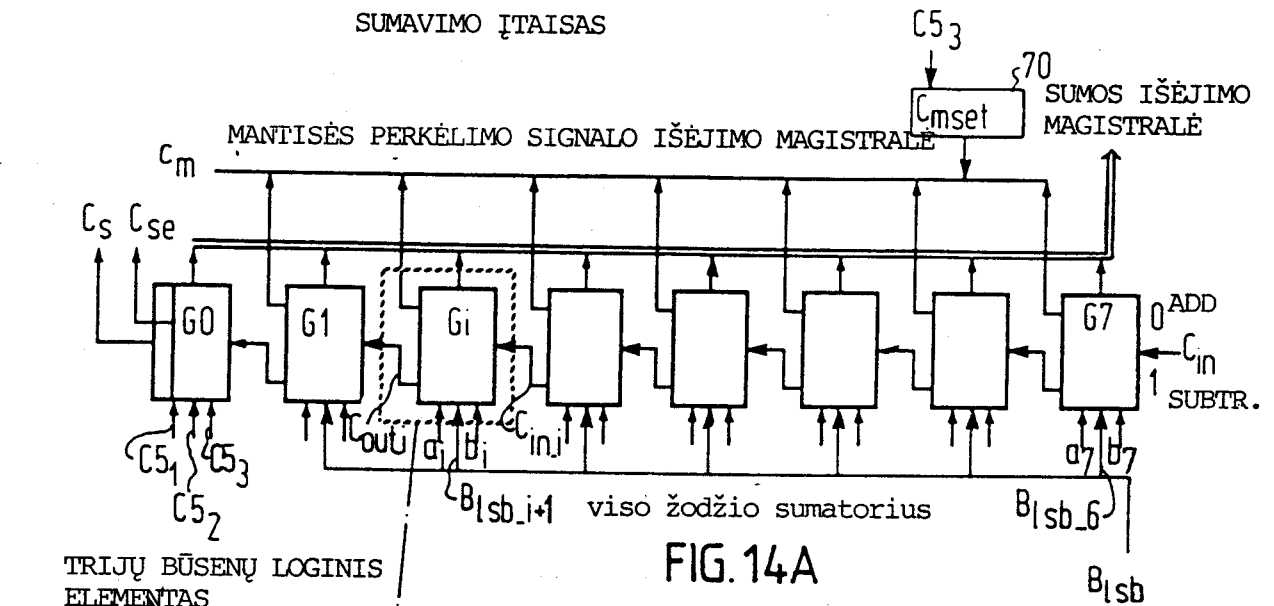


FIG.13

SUMAVIMO ĮTAISAS



15/15

