



ÚŘAD PRO VYNÁLEZY
A OBJEVY

POPIS VYNÁLEZU K AUTORSKÉMU OSVĚDČENÍ

235602

(11) (B1)

(51) Int. Cl.³

G 11 C 15/00

/22/ Přihlášeno 14 07 83

/21/ /PV 5321-83/

(40) Zveřejněno 17 09 84

(45) Vydáno 15 11 86

(75)

Autor vynálezu

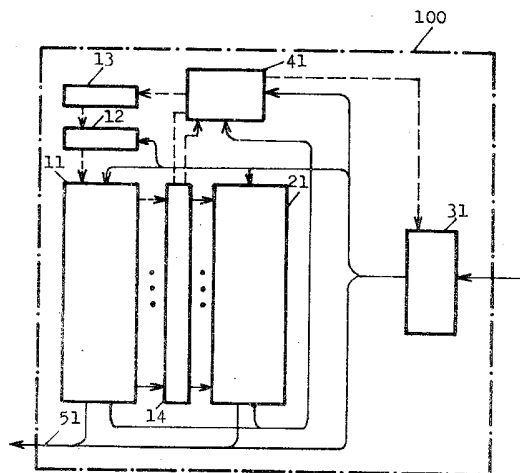
CHLOUBA VÁCLAV ing. CSc., PRAHA

(54) Paměť číslicového počítače řízeného tokem dat

Vynález se týká počítačů s novou architekturou, zaměřených na dosažení vysokého výkonu paralelním prováděním operací.

Podstatou vynálezu je uspořádání paměti číslicového počítače řízeného tokem dat. Paměť je rozdělena na část asociativní a paměťovou. Výběr instrukčních buněk podle obsahu umožňuje vysokou rychlost předávání do bloku paralelně pracujících procesorů.

Oblastí využití jsou výkonné číslicové počítače, zejména pro vědecké a technické výpočty.



Obr. 4

Vynález se týká paměti číslicového počítače řízeného tokem dat.

Vývoj číslicových počítačů se vyznačuje stále rostoucími požadavky na jejich výkon. Protože zvyšování pracovních kmitočtů je omezeno jak rychlostí logických prvků, tak i konečnou rychlostí šíření signálů na spojovacích vedeních, je k dispozici - kromě použití zcela nových fyzikálních principů - prakticky jediný způsob, jak zvýšit výkon počítačů, a to paralelním prováděním funkcí. Toto paralelní provádění lze realizovat na několika úrovních a tomu odpovídá i řešení struktury systému:

a/ Řetězový procesor, který provádí operaci nebo skupinu operací, rozdělených do řady elementárních kroků, v časovém překrytí na celém bloku dat. Jde o jakousi analogii pásové výroby.

b/ Specializovaný maticový procesor, který provádí jeden sled operací na velkém souboru dat současně.

c/ Multiprocessorový systém vybavený několika procesory, z nichž každý řeší nezávisle buď jednu z několika úloh, nebo jednu z několika samostatných částí jedné úlohy, zatímco ostatní části úlohy se řeší v ostatních procesorech a podobně.

Každé z těchto řešení má své výhody a nevýhody. Multiprocessorový systém je plně univerzální, ovšem zvýšení výkonu je - vzhledem k vedlejším činnostem, souvisejícím s organizací práce systému - menší, než by odpovídalo součinu výkonu jednoho procesoru a počtu procesorů. Řetězový procesor je sice výkonnější, ale jeho skutečný výkon silně závisí na charakteru řešených úloh. Společnou nevýhodou obou řešení je nízký stupeň opakovaného použití polovodičových prvků velké integrace - s výjimkou prvků paměťových - což je nevýhodné jak z hlediska spolehlivosti, tak i ekonomie ve výrobě, rozměrů, příkonu a dalších parametrů.

Nejomezenější z hlediska použitelnosti je maticový procesor, který je vhodný pouze pro řešení úloh určitého typu.

Z této situace vyplynuly snahy o nalezení nových architektur, zásadně se lišících od klasického řešení, připisovaného zpravidla von Neumannovi, jež by umožnily stupňování paralelní činnosti, a tím i výkonu číslicových počítačů při současném využití všech výhod vyplývajících z použití polovodičových prvků velké a rozsáhlé integrace. Jedním z těchto perspektivních způsobů řešení systému číslicového počítače jsou takzvané počítače řízené tokem dat, které lze charakterizovat těmito základními vlastnostmi:

- volitelný počet shodných procesorů pracujících paralelně,
- spouštění každé operace ihned, jakmile jsou k dispozici její operandy,
- nevyžaduje operační systém - který u dnešních počítačů snižuje užitečný výkon až i na méně než polovinu - s výjimkou operačního systému případného minipočítače obsluhujícího vstupy a výstupy.

Sériový tvar programů dnešních počítačů není ovšem pro počítače řízené tokem dat vhodný, protože neumožňuje vyjádřit možné paralelnosti provádění. Vhodnějším způsobem reprezentace programů toku dat jsou orientované grafy, jejichž uzly zobrazují operace a hrany přenosy proměnných mezi operacemi.

Uložení programu ve tvaru orientovaného grafu však vyžaduje novou organizaci paměti počítače řízeného tokem dat. Ze způsobu činnosti takového počítače vyplývá, že zde lze s výhodou využít principu asociativní paměti.

Předmětem tohoto vynálezu je paměť počítače řízeného tokem dat, obsahující asociativní část s dotazovým a maskovacím registrem a blokem paměti shody, dále paměťovou část, vyrovnávací paměť a řídicí procesor, přičemž výstup vyrovnávací paměti je spojen se vstupem dotazovací paměti.

vaného registru, vstupem asociativní části, vstupem paměťové části, vstupem řídicího registru a výstupní sběrnici, která je také spojena s výstupem asociativní části a s výstupem paměťové části, zatímco další výstup asociativní části a výstup paměťové části jsou spojeny s dalšími vstupy řídicího procesoru, přičemž jeden z výstupů řídicího procesoru je spojen se vstupem maskovacího registru.

Hlavní výhodou tohoto řešení je rozdělení paměti ve směru délky slova na dvě části, z nichž jen jedna, rozměrově menší, je vybavena schopností asociativního výběru, zatímco druhá část je tvořena standardní pamětí s libovolným výběrem. Tímto uspořádáním se dosahuje podstatné úspory proti řešení s celou pamětí asociativní. Současný stav technologie polovodičových integrovaných pamětí realizaci potřebných prvků s požadovanými parametry kapacity a rychlosti při přijatelných nákladech umožňuje.

Podstata vynálezu bude zřejmá z dalšího popisu s použitím výkresů, kde na obr. 1 je znázorněn příklad úseku programu pro počítač řízený tokem dat, obr. 2 znázorňuje základní strukturu počítače řízeného tokem dat, obr. 3 obsahuje příklad uspořádání polí v buňkách paměti podle vynálezu a obr. 3b příslušný formát hodnot na vstupu této paměti, na obr. 4 je blokové schéma paměti podle vynálezu pro počítač řízený tokem dat, na obr. 5 je program v paměti podle vynálezu odpovídající příkladu na obr. 1, obr. 6 je blokové schéma zdokonaleného uspořádání paměti podle vynálezu pro počítač řízený tokem dat.

Na obr. 1 je znázorněn jednoduchý příklad úseku programu pro vyčíslení algebraického výrazu. Graf obsahuje dva typy uzlů: uzly 5, 6, 8 a 7, 10 aritmetických operací násobení a sčítání, každý se dvěma vstupy a uzly 1, 2, 3, 4 prázdné operace NO, jejíž funkcí je pouze zdvojení hodnoty, přicházející na její jediný vstup. Zvláštním případem je uzel 9 násobení konstantou s jedním vstupem.

Výpočet probíhá zcela asynchronně, určující jsou pouze okamžiky příchodu proměnných na vstupy uzlů a ovšem doba potřebná k provedení jednotlivých operací. Je zřejmé, že systém nepotřebuje žádnou synchronizaci mezi bloky ani ústřední řadič.

Jednoduchý příklad podle obr. 1 umožňuje provádět v každém okamžiku současně dvě operace, čili doba výpočtu se zkracuje přibližně na polovinu. U složitějších výpočtů může být toto zkrácení doby výpočtu - podle charakteru úlohy - větší nebo menší. Pokud je k dispozici dost úloh nebo větví jedné úlohy, aby byly všechny procesory stále zaměstnány, je celkový výkon dán součinem počtu procesorů a výkonu jednoho procesoru, neboť předpokládáme vzájemně shodné procesory.

Operace v příkladu na obr. 1 jsou ovšem jen zlomkem operačního kódu procesorů. Kromě základních aritmetických a logických operací lze uvažovat i další operace jako mocniny a odmocniny a funkce, zejména goniometrické a další.

Základní struktura počítače řízeného tokem dat, uvažovaná v tomto vynálezu, je znázorněna na obr. 2. Obsahuje paměť 100, blok 20 vzájemně shodných procesorů, distribuční síť 30, rozhodovací síť 40 a přepínač 50.

Funkci této struktury lze stručně popsat takto: V paměti 100 jsou uloženy programy ve tvaru orientovaných grafů. Jakmile některý uzel dostane všechny své vstupní hodnoty, tj. operandy, paměť 100 vyšle příslušnou instrukci do distribuční sítě 30, která ji zavede do jednoho z volných procesorů v bloku 20 procesorů. Rozhodovací síť 40 řadí výsledky operací, probíhající v bloku 20 procesorů, do společného vedení. Výsledky postupují přes přepínač 50 do paměti 100, kde se ukládají jako vstupy do příslušných uzlů a tak dále. Přepínač 50 umožňuje jednak vkládání nových programů a vstupních hodnot do paměti 100, jednak vysílání výsledků z bloku 20 procesorů na výstup systému.

Formát instrukcí v paměti 100 podle vynálezu je naznačen na obr. 3a. Každá instrukční

buňka se skládá ze dvou částí: asociativní a paměťové. Asociativní část obsahuje například tato čtyři pole: 101 - označení programu /uživatele/, 102 - číslo uzlu programu, 103 - příznak platnosti hodnoty a označení vstupu uzlu, pro který je hodnota určena, 104 - index. Pole 101 umožňuje, aby v paměti mohlo být současně několik různých programů, které se vzájemně neovlivňují, pole 104 se uplatní při opakovaném použití části programu v průběhu výpočtu, například při iteračních výpočtech nebo při zpracování složitějších datových struktur, jako jsou matice nebo vektory. Index zabezpečuje, aby do uzlu přišly vždy jen hodnoty, které k sobě patří. Uzly, které jsou součástí smyčky v programu, jsou odlišeny příznakem v poli 211 operačního znaku.

Paměťová část instrukční buňky obsahuje tato pole: 211 - operační znak; určuje, která operace se má provést, 212 - číslo následujícího uzlu programu a označení vstupu, na který se má přenést výsledek operace, případně příznak výstupu, má-li se hodnota vyslat přepínačem 50 na výstup, 213 - hodnota.

Poznámka. Je výhodné, aby pole 212 mohlo obsahovat označení například dvou uzlů a jejich vstupů, protože tím se umožní větvení programu v každém uzlu nebo současně vyslání hodnoty na výstup bez větší ztráty času.

Popis v předchozích odstavcích odpovídá uzlům, reprezentujícím operace se dvěma vstupy. Řada operací však pracuje jen s jedním vstupem. Tyto operace mají zvláštní příznak v poli 211, příznak platnosti v poli 103 je trvale II a pole 213 se nevyužívá. Zvláštním případem jsou operace s konstantou, opět odlišené příznakem v poli 211, u nichž je příznak v poli 103 také trvale II, přičemž v poli 213 je uložena příslušná konstanta.

Na obr. 3a jsou vyčerněním levého horního rohu znázorněna pole, jejichž obsah se v průběhu řešení úlohy mění. Obsah ostatních polí zůstává po celou dobu řešení úlohy beze změny.

Na obr. 3b je znázorněn formát výsledků vystupujících z bloku 20 procesorů a procházejících přes rozhodovací síť 40 a přepínač 50 na vstup paměti 100. Jak je zřejmé, výsledky obsahují pět polí: 301 - označení programu, 302 - číslo uzlu programu, do něhož výsledek přichází jako vstupní hodnota, 303 - označení vstupu tohoto uzlu, 304 - index, 305 - hodnota.

Poznámka. Na obr. 3a a 3b jsou vynechána pole redundantních informací, určených k zabezpečení proti chybám. Obdobně se neuvažují další pomocná pole v instrukčních buňkách umožňující například vstup přídatných podmínek do uzlů programu. Tato pole nemají vliv na podstatu vynálezu.

Jak je zřejmé z obr. 4, paměť 100 podle vynálezu sestává z asociativní části 11, vybačené dotazovým registrem 12, maskovacím registrem 13 a blokem 14 paměti shody a dále paměťové části 21, vyrovnávací paměti 31 a řídicího procesoru 41. Funkci paměti 100 podle vynálezu můžeme popsat takto: Předpokládáme, že v paměti 100 je už uložen program v podobě orientovaného grafu. Když se na vstupu paměti 100 objeví nová hodnota, ať už přicházející prostřednictvím přepínače 50 z okolí nebo jako výsledek z bloku 20 procesorů přes rozhodovací síť 40, vloží se nejprve do vyrovnávací paměti 31. Z obsahu polí 301, 302 a 304 řídicí procesor 41 vytvoří klíč, který vloží do dotazového registru 12 a příslušnou masku do maskovacího registru 13. Pak proběhne asociativní porovnání klíče s obsahem asociativní části 11, jehož cílem je vyhledání uzlu, do něhož má přicházející hodnota vstoupit jako operand. Tento uzel je charakterizován tím, že v polích 101, 102 a 104 má tentýž obsah, jaký je v polích 301, 302 a 304 hodnoty nyní uložené ve vyrovnávací paměti 31. Za předpokladu, že takový uzel /instrukční buňka/ v paměti 100 je, řídicí jednotka 41 zjistí stav pole 103 nalezené buňky se zřetelem k obsahu polí 211 a 303. Mohou nastat tyto případy:

aa/ Nalezená buňka představuje uzel s jediným vstupem nebo uzel operace s konstantou. V tomto případě se celá instrukční buňka - kromě obsahu pole 103 - spolu s obsahem pole 305 hodnoty ihned vyšle na výstupní sběrnici 51 a přes distribuční síť 30 k provedení v některém

z volných procesorů v bloku 20 procesorů - viz obr. 2. Při vyslání instrukční buňky se obsah pole 103 nemění.

ab/ Nalezená buňka představuje uzel se dvěma vstupy a příznak v poli 103 signalizuje, že jedna ze vstupních hodnot je již k dispozici v poli 213 paměťové části. Také v tomto případě se celá instrukční buňka - kromě obsahu pole 103 - spolu s obsahem pole 305 vyšle z paměti 100 výstupní sběrnici 51 k provedení. Současně se obsah pole 103 vynuluje.

ac/ Nalezená buňka představuje uzel se dvěma vstupy, avšak podle příznaku v poli 103 je pole 213 prázdné, takže vyrovnávací paměť 31 nyní obsahuje teprve první z obou vstupních operandů uzlu. V tomto případě se obsah pole 305 zapíše do pole 213 a podle obsahu pole 303 se opraví příznak v poli 103. Na výstupní sběrnici 51 se nic nevysílá.

Jestliže hledání v asociativní části 11 paměti 100 nevedlo k nalezení uzlu, tj. žádná z instrukčních buněk nesplňuje podmínku shody obsahu odpovídajících si polí, přicházejí v úvahu tyto příčiny:

ba/ Pro dané označení programu v polích 101, 301 neexistuje v paměti 100 číslo uzlu programu 102 shodné s obsahem pole 302. Příčinou je pravděpodobně chyba v programu.

bb/ Program s označeními obsažených v poli 301 v paměti 100 není. Příčinou je nejspíše chyba operátora, který nezavedl program do paměti; předpokládáme, že jak obsah paměti 100, tak i přicházející hodnoty jsou zabezpečeny kontrolními bity a že tedy obsah pole 301 nemůže být chybný.

bc/ Pro dané označení programu v polích 101, 301 a číslo uzlu v polích 102, 302 není v paměti 100 uzel, jehož index v poli 104 odpovídá obsahu pole 304, což znamená, že přicházející hodnota nepatří do nejbližší příštího cyklu provedení tohoto uzlu, ale až do některého dalšího. Pro tento případ má vyrovnávací paměť 31 několik odkládacích míst, kde se tato hodnota podrží. Současně se případ zaregistruje v paměti řídicího procesoru 41, který po každé opravě indexu v poli 104 příslušné instrukční buňky daného programu zkusí, zda se odložená hodnota již může použít. Protože případy, kdy hodnoty přijdou v jiném pořadí než jsou zapotřebí, jsou obecně málo pravděpodobné, stačí ve vyrovnávací paměti 31 rezervovat jen několik míst, odhadem například 1 promile počtu míst v částech 11 a 21 paměti 100.

bd/ Jiný chybový stav nastane, jestliže příznak v poli 103 signalizuje určení hodnoty v poli 213 pro týž vstup jako pole 303 pro hodnotu v poli 305. Také v tomto případě jde o chybu v programu.

Případy ba, bb a bd zjišťuje řídicí procesor 41 a hlásí obsluhu. Aby chyba nezpůsobila zastavení všech programů uložených v paměti 100, odloží se hodnota, která vyvolala chybový stav, do některého rezervního místa ve vyrovnávací paměti 31.

Pro ilustraci je uveden na obr. 5 obsah paměti 100 podle vynálezu pro program pro výpočet podle obr. 1. Program obsahuje celkem deset uzlů, z toho čtyři uzly jsou pomocné prázdné operace, tři uzly představují operaci násobení, dva uzly sčítání a jeden uzel násobení konstantou. Pokud se nepředpokládá opakované použití programu ve smyčce, zůstanou indexová pole 104 a 304 prázdná. Před spuštěním programu jsou prázdná i všechna pole 213 hodnot. Příchodem vstupních hodnot x a y , jejichž formát je znázorněn na obr. 5 dole, se program spustí. Vstupní hodnoty x a y se nejprve provedením uzlů 1, 2, 3 a 4 ztrojí a objeví se jednak jako vstupní hodnoty uzlů 5 a 6 - viz obr. 1 - jednak jako vstupní hodnoty uzlu 8. Zatímco výstupy uzlů 5 a 6 vstoupí do uzlu 7, výstup z uzlu 8 se objeví na vstupu uzlu 9 násobení konstantou. Výstup z uzlu 7 a výstup z uzlu 9 spolu spustí závěrečný uzel 10, jehož výsledek se vyšle na výstup.

Operace, odpovídající jednotlivým uzlům programu, probíhají zcela nezávisle a asyn-

chronně. Každá se spustí, jakmile jsou k dispozici všechny její vstupní hodnoty, operandy. Jak plyne z grafu na obr. 1 i z popisu, v tomto jednoduchém případě probíhá výpočet paralelně ve dvou větvích, čili doba výpočtu je asi poloviční, než by byla při sekvenčním výpočtu stejně rychlým konvenčním procesorem.

Vyjde-li se ze současného stavu technologie polovodičových integrovaných obvodů, může se odhadnout doba zpracování jedné hodnoty, tj. vyhledání příslušného uzlu v paměti 100 podle vynálezu a vyslání instrukční buňky sběrnici 51 asi na 0,5 až 1 μ s, zatímco průměrná doba operace procesoru v bloku 20 procesorů bude asi 4 μ s. Z toho plyne, že paměť 100 může obsloužit asi čtyři procesory. Při požadavku vyššího celkového výkonu je možno uspořádání zdokonalit podle obr. 6, kde paměť 100 podle vynálezu obsahuje navíc multiplexor 61, zatímco ostatní části paměti 100, tj. asociativní část 11 s dotazovým registrem 12, maskovacím registrem 13 a blokem 14 paměti shody, dále paměťová část 21 a vyrovnávací paměť 31, jsou rozděleny do několika shodných a samostatných bloků, pracujících paralelně. Výstupní sběrnice 51 těchto samostatných bloků jsou zavedeny do distribuční sítě 30, která má na rozdíl od předešlé potřebný počet vstupů a umožňuje současné propojení těchto vstupů s výstupy. Pokud řídící procesor 41 postačí svým výkonem, může být společný pro celou paměť 100 nebo také rozdělen na několik částí, případně v hierarchickém uspořádání.

Multiplexor 61 má za úkol nejprve rozmístit uzly programu i celé programy tak, aby všechny bloky paměti 100 a také procesory v bloku 20 procesorů mohly pracovat současně, aby se všechny prostředky systému využily s cílem dosažení nejvyššího výkonu. Možných způsobů řízení multiplexoru 61 je jistě celá řada, jako nejjednodušší příklad je možno uvést použití čísla uzlu programu modulo n, kde n je počet bloků paměti 100.

Řešení systému řízené tokem dat podle obr. 6 s blokem 20 obsahujícím například 20 procesorů a paměti 100 podle vynálezu, rozdělenou do pěti až osmi paralelně pracujících bloků, umožňuje při dnešním stavu technologie polovodičových integrovaných prvků dosáhnout výkonu řádu 5 M operací/s.

Kromě vysokého výkonu vykazuje systém podle obr. 2 a 6 výhodu stavebnicovosti, tj. možnosti vytvářet různé výkonné systémy volbou počtu procesorů v bloku 20 procesorů a volbou počtu bloků v paměti 100, aniž by bylo nutno jakkoli upravovat programové prostředky. Další výhodou je opakované použití celé řady shodných bloků, jak procesorů, paměťových i dalších, což umožňuje využití výhod prvků velké, případně rozsáhlé integrace, a tedy ekonomickou realizaci celého systému s dalšími příznivými důsledky pro spolehlivost, příkon, objem a podobně.

Další výhodou řešení podle vynálezu je možnost v případě poruchy některého bloku paměti 100 nebo některého procesoru v bloku 20 procesorů tento blok, případně procesor zásahem do řízení multiplexoru 61, respektive distribuční sítě 30 vyřadit z přidělování a pokračovat v provozu sice s poněkud sníženými parametry, ale bez nutnosti jakkoli měnit řešené programy.

Na schématech na obr. 4 a 6 jsou pro přehlednost znázorněna jen ta vedení, která jsou důležitá z hlediska podstaty vynálezu. Vynechána jsou také hradla na těchto vedeních, kterými řídící procesor 41 řídí přenosy mezi jednotlivými bloky uvnitř paměti 100.

Konkrétní realizace paměti 100 podle vynálezu bude vyžadovat obvody pro kontrolu správnosti vstupních i výstupních hodnot. Ani tyto obvody přímo nesouvisejí s podstatou vynálezu, a proto jsou na schématech na obr. 4 a 6 pro přehlednost také vynechány.

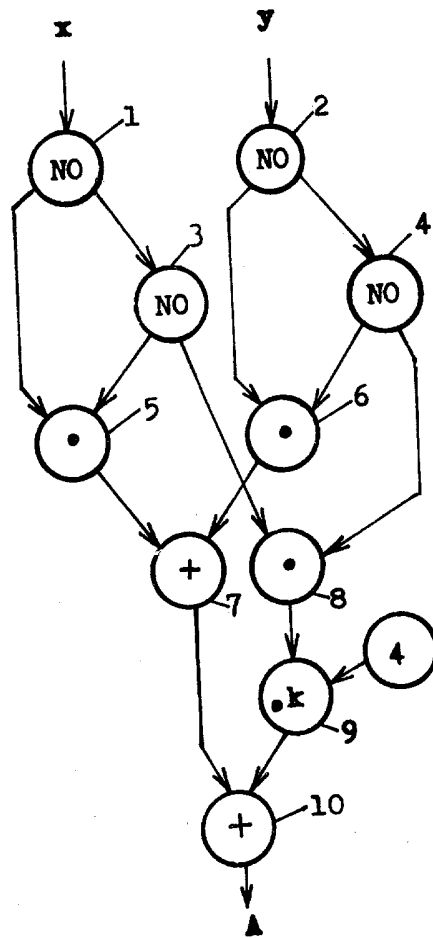
PŘEDMĚT VYNÁLEZU

1. Paměť číslicového počítače, řízeného tokem dat, vyznačená tím, že výstup vyrovnávací paměti /31/ je spojen se vstupem dotazového registru /12/, vstupem asociativní části /11/, vstupem paměťové části /21/, vstupem řídicího procesoru /41/ a s výstupní sběrnicí /51/, která je také spojena s výstupem asociativní části /11/ a s výstupem paměťové části /21/, zatímco další výstup asociativní části /11/ a další výstup paměťové části /21/ jsou spojeny s dalšími vstupy řídicího procesoru /41/ a jeden z výstupů řídicího procesoru /41/ je spojen se vstupem maskovacího registru /13/.

2. Paměť podle bodu 1, vyznačená tím, že asociativní část /11/ a paměťová část /21/ jsou rozděleny do stejného počtu shodných bloků, kde každá asociativní část /11/ má svůj dotazový registr /12/, maskovací registr /13/ a blok /14/ pamětí shody a každý z těchto shodných bloků má vlastní vyrovnávací paměť /31/ a samostatnou výstupní sběrnicí /51/, přičemž vstupy vyrovnávacích pamětí /31/ jsou jednotlivě spojeny s výstupy multiplexoru /61/ ovládaného vhodnou částí vstupujících informací.

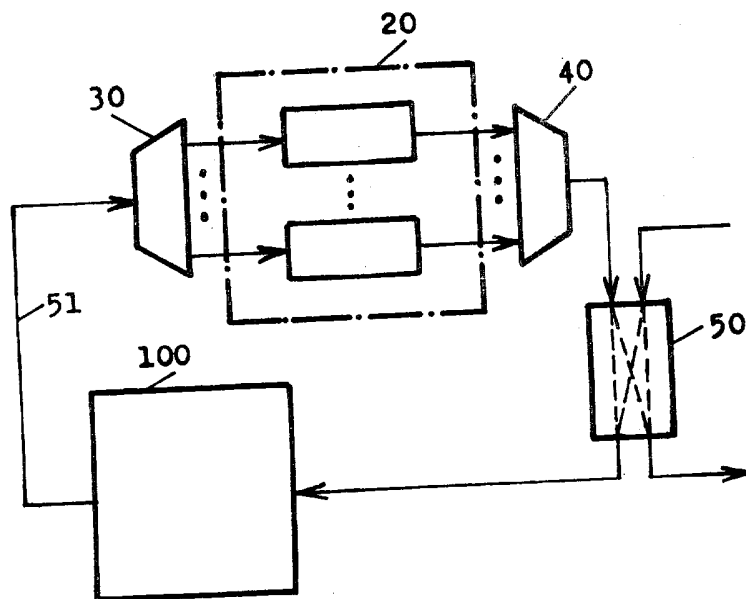
3 výkresy

235602

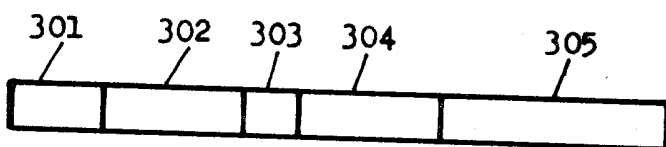
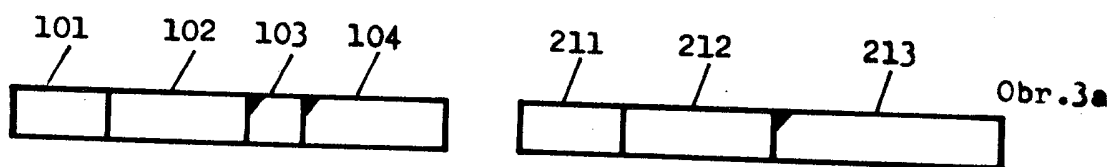


$$A = x^2 + 4xy + y^2$$

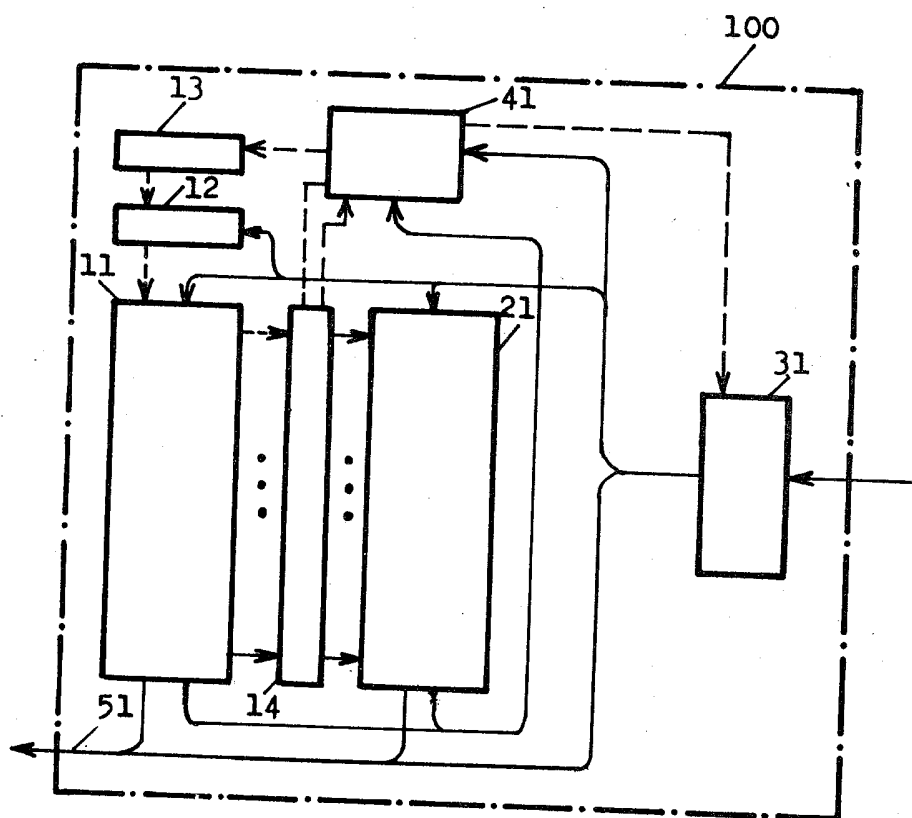
Obr. 1



Obr. 2



Obr. 3b



Obr. 4

101	102	103	104	211	212	213
135	1	II	-	NO	3I,5I	-
135	2	II	-	NO	4I,6I	-
135	3	Ø	-	NO	5II,8I	-
135	4	Ø	-	NO	6II,8II	-
135	5	Ø	-	.	7I	-
135	6	Ø	-	.	7II	-
135	7	Ø	-	+	10I	-
135	8	Ø	-	.	9I	-
135	9	II	-	.K	10II	4
135	10	Ø	-	+	out	-

135	1	I	-	x
135	2	I	-	y
301	302	303	304	305

Obr. 5

