



ÚŘAD PRO VYNÁLEZY
A OBJEVY

POPIS VYNÁLEZU K AUTORSKÉMU OSVĚDČENÍ

215264

(11) (B1)

(22) Přihlášeno 28 11 77
(21) (PV 7833-77)

(51) Int. Cl.³
G 06 F 9/10

9/30 opr.

(40) Zveřejněno 29 08 80

(45) Vydáno 30 04 84

(75)

Autor vynálezu

MAGERSKY PETKO STEFANOV ing. CSc., SOFIA (BLR),
MIRTES BOHUMIL ing. CSc., PRAHA (ČSSR)

(54) Zapojení pro zpracování informace

Vynález se týká zapojení pro zpracování informace, sestavené z centrálního procesoru, z pevné paměti mikroinstrukcí a z bloku pro adresový výběr.

Je známo zapojení pro zpracování informace, které sestává z centrálního procesoru, jehož vstup je spojen s výstupem pevné paměti mikroinstrukcí, jejíž druhý výstup je spojen s blokem pro adresový výběr, jehož vstup je spojen se vstupem pevné paměti mikroinstrukcí. Daný sled mikroinstrukcí se zpracovává postupným vybavováním mikroinstrukcí, které se pak vykonají centrálním procesorem. Mikroinstrukce se nahrávají do pevné paměti mikroinstrukcí a vybírají podle adresy připravené v bloku pro adresový výběr. Od okamžiku zadání adresy na vstup bloku pro adresový výběr po vybavení odpovídající mikroinstrukce uběhne určitá doba, která se označuje jako ROM cyklus a která představuje součet zpoždění bloku adresového výběru (t_{adresa}) a zpoždění při vybavování mikroinstrukce z pevné paměti mikroinstrukcí ($t_{\text{vybavení}}$):

ROM cyklus = $t_{\text{adresa}} + t_{\text{vybavení}}$ (ns).

V případech, kdy je doba provedení mikroinstrukce centrálním procesorem označována jako CP cyklus menší než ROM cyklus, bude centrální procesor nucen v každém taktu čekat na vybavování příští mikroinstrukce z pevné paměti mikroinstrukcí. To vede k časové ztrátě v práci

centrálního procesoru a tím i celého zařízení.

Nedostatek popsaného zařízení spočívá ve zmiňované ztrátě čekáním centrálního procesoru; tato ztráta je tím větší, čím je rozdíl obou cyklů větší.

Je známo druhé zapojení pro zpracování informace, které odstraňuje toto nežádoucí zpoždění rychlejšího centrálního procesoru paralelním vybavením „k“ operačních částí „k“ jdoucích po sobě mikroinstrukcí z jedné adresy pevné paměti mikroinstrukcí, kde „k“ představuje

$k \cong \frac{\text{ROM cyklus}}{\text{CP cyklus}}$

a $\cong$ je symbol označující nejbližší větší zaokrouhlenou hodnotu podílu.

Zapojení sestává z centrálního procesoru, jehož vstup je spojen s výstupem bloku pro paralelní zpracování, jehož vstup je spojen s výstupem pevné paměti mikroinstrukcí a jehož druhý výstup — se vstupem bloku pro adresový výběr, jehož výstup je spojen se vstupem pevné paměti mikroinstrukcí. Z jedné adresy pevné paměti mikroinstrukcí se vybavuje „k“ operačních částí následujících mikroinstrukcí, které postupují do bloku pro paralelní zpracování; začne-li centrální procesor plnit první operační část první mikroinstrukce, začínají se v soustavě blok pro adresový výběr — pevná paměť mikroinstrukcí připravovat následující „k“ operačních částí mikroinstrukcí. V okamžiku zakončení zpracování poslední z najednou vybavených operačních částí mi-

kroinstrukcí jsou na výstupu pevné paměti mikroinstrukcí připraveny následující „k“ operačních částí mikroinstrukcí, které se stejným způsobem zpracovávají v centrálním procesoru. Centrální procesor pracuje plnou rychlostí a nečeká na pomalejší části soustavy.

Nedostatek druhého zapojení spočívá ve velkém počtu bitů, které se vybavují z jedné adresy pevné paměti mikroinstrukcí. V tomto zapojení je záznam na jedné adrese „k“-krát delší než záznam na jedné adrese z prvního zapojení. Protože obvykle operační části mikroinstrukcí jsou dlouhé několik desítek bitů, se zvýšením „k“ velmi rychle narůstá počet vybavovaných bitů, což vyvolává rychlou expanzi logických obvodů bloku pro paralelní zpracování.

Je známo třetí zapojení pro zpracování informace, které odstraňuje nežádoucí čekání rychlejšího centrálního procesoru při splnění podmínky:

$$t_{\text{vybavení}} < \text{CP cyklus} < \text{ROM cyklus} = t_{\text{adresa}} + t_{\text{vybavení}}$$

Význam podmínky spočívá v tom, že pevná paměť mikroinstrukcí je rychlejší než centrální procesor a jen v zapojení s blokem pro adresový výběr vyvolává tuto nerovnost.

Zapojení sestává z centrálního procesoru, jehož vstup je spojen s výstupem pevné paměti mikroinstrukcí, jejíž druhý výstup je spojen se vstupem bloku pro adresový výběr a zároveň se vstupem bloku pro paralelní zpracování. Výstup bloku pro adresový výběr je spojen s druhým vstupem bloku pro paralelní zpracování, jehož výstup je spojen se vstupem pevné paměti mikroinstrukcí. Zapojení odstraňuje nežádoucí zpoždění centrálního procesoru, když při nucené adresaci a mikroinstrukcích bez rozvětvení propouští adresu příští mikroinstrukce blokem pro paralelní zpracování rovnou do pevné paměti mikroinstrukcí a tímto odstraňuje zpoždění bloku pro adresový výběr a nahrazuje ho menším zpožděním bloku pro paralelní zpracování. Na druhé straně pevná paměť mikroinstrukcí je rychlejší než centrální procesor, stačí ho včas zásobovat mikroinstrukcemi a centrální procesor pracuje maximální rychlostí. Přitom je blok pro adresový výběr volný a může se použít na paralelní přípravu adresy nezávislého rozvětvení sledu mikroinstrukcí, které tímto způsobem proběhne maximální rychlostí.

Charakteristické pro toto zapojení je, že je omezeno požadavkem:

$$t_{\text{vybavení}} < \text{CP cyklus} \quad (\text{ns}) \quad [1] \\ \text{a nelze jej použít, když není tento požadavek splněn a dosáhnout maximální rychlost centrálního procesoru.}$$

Je známo čtvrté zapojení pro zpracování informace, které odstraňuje nežádoucí zpoždění rychlejšího centrálního procesoru pro libovolnou rychlost vybavení pevné paměti mikroinstrukcí. Zapojení sestává z centrálního procesoru, jehož vstup je spojen s výstupem multiplexoru, k jehož stejnému výstupu jsou připojeny paralelně vstupy nejméně dvou bloků pro adresový výběr, jejichž výstupy jsou spojeny se vstupy nejméně dvou pevných pamětí mikroinstrukcí, jejichž výstupy jsou spojeny se vstupy multiplexoru. Zapojení pracuje následovně:

Při obvyklém vybavení nucenou adresací každá mikroinstrukce se skládá z operační části, která řídí činnost operační jednotky, a z adresové části, ukazující adresu příští mikroinstrukce. Toto zapojení zavádí souběžné procesy v současném přečtení z pevné paměti mikroinstrukcí a v současném zpracování „k“ adres mikroinstrukcí pomocí „k“ bloků pro adresový výběr a pomocí odpovídajícího paralelního vybavení těchto mikroinstrukcí z „k“ pevných pamětí mikroinstrukcí tak, že pokud centrální procesor vykonává první „k“ mikroinstrukcí, podle paralelně zpracováváných „k“ adres se připravují současně následující „k“ mikroinstrukcí, které jsou k dispozici ještě než centrální procesor ukončí zpracování prvních „k“ mikroinstrukcí. Tímto je centrální procesor rytmicky zásobován mikroinstrukcemi, nečeká na pomalejší části soustavy a pracuje maximální rychlostí.

Při obvyklém vybavení nucenou adresací každá mikroinstrukce se skládá z operační části, která řídí činnost operační jednotky, a z adresové části, ukazující na adresu příští mikroinstrukce. Zapojení pro zpracování informace podle vynálezu zavádí souběžné procesy v současném přečtení z pevné paměti mikroinstrukcí a v současném zpracování „k“ adres mikroinstrukcí pomocí „k“ bloků pro adresový výběr, podle kterých se pak sériově z pevné paměti mikroinstrukcí rychleji než centrální procesor vybavují postupně „k“ mikroinstrukcí. Tímto způsobem, než centrální procesor stačí zpracovat tyto „k“ mikroinstrukce, se paralelně obměňují adresy následujících „k“ mikroinstrukcí a takto pracuje maximální rychlostí.

Uvedená podstata se nemění, jestliže adresa příští mikroinstrukce není částí vybavené mikroinstrukce.

Je zřejmé, že i zde informace uložená na jedné adrese v pevné paměti mikroinstrukcí má větší počet bitů, neboť na jedné adrese se kromě vlastní informace, tj. mikrooperační plus adresová část příští mikroinstrukce, musí nacházet ještě „k-1“ adres příštích mikroinstrukcí. Pokud se ale v již dříve popsaném druhém zapojení na jedné adrese v pevné paměti mikroinstrukcí nacházejí „k“ operačních částí a každá je délky obvykle několik desítek bitů, to zde se opakuje „k“-krát adresové části a každá adresová část je kratší 10 bitů.

Příklad: Mikroinstrukce emulované PDP 11/40 je dlouhá 54 bitů, ze kterých 45 je operačních a 9 je adresových. Je-li podstata pevné paměti mikroinstrukce taková, že $k > 1$, pak je počet bitů na jedné adrese v pevné paměti mikroinstrukcí druhého zapojení $(54 + 45.k)$ a počet bitů na jedné adrese v pevné paměti mikroinstrukcí v zapojení podle vynálezu je $(45 + 9.k)$. Pro $k = 2$ obdržíme 144 bitů a 63 bitů.

Uvedené nedostatky prvního a druhého zapojení odstraňuje zapojení pro zpracování informace, sestavené z centrálního procesoru, z pevné paměti mikroinstrukcí a z bloku pro adresový výběr podle vynálezu, jehož podstatou je, že obsahuje nejméně dva bloky pro adresový výběr, jejichž výstupy jsou spojeny se vstupy multiplexoru, jehož výstup je spojen se vstupem pevné paměti mikroinstrukcí, jejíž výstup je spojen se vstupem

centrálního procesoru a se vstupy bloků pro adresový výběr.

Zapojení pro zpracování informace podle vynálezu představuje variantu uvedeného čtvrtého zapojení pro zpracování informace v případě, že je pevná paměť mikroinstrukcí rychlejší ve své činnosti než centrální procesor a také zajišťuje maximální rychlost centrálního procesoru.

Zapojení pro zpracování informace podle vynálezu představuje další možnost, další varianta uvedeného třetího zapojení pro zpracování informace platí jen v případě rychlejší pevné paměti mikroinstrukcí než centrální procesor a také zajišťuje maximální rychlost centrálního procesoru.

Podrobněji je zapojení pro paralelní zpracování informace, podle uvedeného vynálezu, vysvětleno pomocí výkresu, který představuje blokové schéma zapojení.

Zapojení pro zpracování informace sestává z centrálního procesoru 1, jehož vstup je spojen s výstupem pevné paměti 6 mikroinstrukcí, který výstup je také spojen se vstupy nejméně dvou bloků 3, 4, ...5 pro adresový výběr, jejichž výstupy jsou spojeny se vstupy multiplexoru 2, jehož výstup je spojen se vstupem pevné paměti 6 mikroinstrukcí.

Pro lepší porozumění je nutno předpokládat, že soustava centrální procesor — pevná paměť mikroinstrukcí — blok pro adresový výběr se chová synchronně, tj. že centrální procesor se nemůže asynchronně zastavovat a čekat na ostatní části soustavy.

Zapojení pracuje, když platí $k > 1$ a podmínka [1], a to následujícím způsobem:

Při spouštění se do bloku 3 pro adresový výběr podává adresa první mikroinstrukce, do bloku 4 pro adresový výběr — adresa druhé, ..., do bloku 5 pro adresový výběr a adresa „ $k-1$ “ taktech zpracovávají. Potom se multiplexor 2 přepíná na výstup bloku 3 pro adresový výběr a zadává první připravenou adresu do pevné paměti 6 mikroinstrukcí. Během jednoho taktu se z pevné paměti 6 mikroinstrukcí vybaví první mikroinstrukce, protože platí podmínka [1], jejíž operační část postupuje do centrálního procesoru 1 a „ k “ adresových částí — do bloků 3, 4, ...5 pro adresový výběr. První adresa se začne hned zpracovávat v bloku 3 pro adresový výběr tak, aby po „ $k-1$ “ taktech byla adresa „ $k+1$ “ — ní mikroinstrukce hotová k podání multiplexorem 2 do pevné paměti 6 mikroinstrukcí. Během zpracování první mikroinstrukce centrálním procesorem 1 se bloku 4 pro adresový výběr přepnutým k výstupu multiplexorem 2 zadává adresa druhé mikroinstrukce do pevné paměti 6 mikroinstrukcí, která se začne vybavovat a je hotová než centrální procesor ukončí zpracování první mikroinstrukce. Multi-

plexor 2 se cyklicky přepíná a na každém „ k “ — taktu je spojen s výstupem bloku 3 pro adresový výběr.

Centrální procesor je tímto rytmicky zásobován mikroinstrukcemi a pracuje bez čekání na zbývající části soustavy. Další předností je to, že na základě podmínky [1] bylo možno zintegrovat jednotlivých „ k “ pevných pamětí mikroinstrukcí z popsaného čtvrtého zapojení pro zpracování informace do jedné pevné paměti mikroinstrukce a tím bez modifikace mikroprogramů snížit „ k “ — krát počet paměťových buněk.

V dalším bude uvedena modifikace popsaného zapojení, které má reálnou praktickou aplikaci, nemění princip činnosti, klade větší požadavky při návrhu mikroprogramů a vede ke značnému zjednodušení uvedeného zapojení.

Na základě algoritmu činnosti a odpovídajícího mikrokódu centrálního procesoru návrhář vytváří mikroprogramovou strukturu činnosti mikroprogramové synchronní soustavy. Tyto mikroprogramy sestávají z větví, ve kterých po každé mikroinstrukci následuje jen jedna mikroinstrukce, z nezávislých a závislých rozvětvení. Po každé mikroinstrukci, ve které vzniká závislé rozvětvení, se dodávají „ $k-1$ “ prázdných mikroinstrukcí NOP (NO OPERATION). Potom se přistupuje k modifikaci takto vytvořené struktury následovně:

a) každá větev mikroprogramů musí obsahovat takový počet mikroinstrukcí, který představuje celý násobek stupně „ k “. Jestliže tomu tak není, pak se přidávají prázdné mikroinstrukce NOP do dosažení nejmenšího celého násobku;

b) podmínky, vyvolávající nezávislá rozvětvení se zadávají „ k “ taktů před okamžikem provedení nezávislého rozvětvení.

Pak v zapojení podle výkresu vznikají následující zjednodušení: kromě bloku 3 pro adresový výběr se všechny zbývající bloky 4, ...5 pro adresový výběr modifikují a stávají se obyčejnými registry. Všechna rozvětvení se provádějí jenom v bloku 3 pro adresový výběr, který zachovává svou úplnou strukturu a možnosti.

Protože se bloky pro adresový výběr v úplné nebo modifikované podobě mohou vyrábět se třístavovými výstupy, při použití se potom jejich výstupy dají spojit paralelně. Tím se multiplexor 2 značně zjednoduší a jeho úloha pak spočívá v adresování výběrových vstupů jednotlivých bloků pro adresový výběr.

Toto zapojení může najít uplatnění ve výpočetní technice jak v jednotném systému počítačích strojů, tak v jednotném systému malých počítačích strojů, při konstruování mikroprogramových systémů, v mikroprocesorové technice, v řadičích periferních zařízení, v nestandardních číslicových zařízeních a v NC strojích.

PŘEDMĚT VYNÁLEZU

Zapojení pro zpracování informace sestavené z centrálního procesoru, z pevné paměti mikroinstrukcí a z bloku pro adresový výběr, který může představovat i registr, vyznačující se tím, že obsahuje nejméně dva bloky (3, 4, 5) pro adre-

sový výběr, jejichž výstupy jsou spojeny se vstupy multiplexoru (2), jehož výstup je spojen se vstupem pevné paměti (6) mikroinstrukcí, jejíž výstup je spojen se vstupem centrálního procesoru (1) a se vstupy bloků (3, 4, 5) pro adresový výběr.

1 výkres

