

VME VODILO V VEČRAČUNALNIŠKIH ARHITEKTURAH

M. COLNARIČ, I. ROZMAN, B. PREMZEL
TEHNIŠKA FAKULTETA MARIBOR

POVZETEK - Članek obravnava problematiko vodil v večprocesorskih sistemih s poudarkom na VME vodilu. To je prirejeno za prioritetni in Round Robin arbitracijski algoritem z Daisy - Chain povezavo na posameznih nivojih. Ker je v novejših raziskavah utemeljeno, da tovrstni algoritem ne daje optimalnih rezultatov, so v članku nakazani možni načini realizacije drugih izbirnih pravil.

ABSTRACT - In the article multiprocessor systems' buses are discussed with accent on the VME bus. It is designed for priority and Round Robin arbitration scheme with Daisy - Chained modules on each level. Since it is proved in recent research that this kind of algorithms are not optimal, some other arbitration protocols are described.

UVOD

Z razvojem mikroprocesorjev prevzemajo mikroročunalniški sistemi čedalje kompleksnejše naloge. Marsikaterih zahtev pa tudi najmočnejši procesorji ne morejo zadovoljiti. Iz tega razloga in ker jim cena nenehno upada so čedalje zanimivejši sistemi, kjer si več procesorjev deli globalne resurse.

Najpomembnejši med njimi je vodilo za prenos podatkov. Preko njega procesorji komunicirajo z drugimi resursi in med seboj. Vsak sistem z več procesorji mora reševati problem dodeljevanja vodila uporabnikom. Uspešnost reševanja tega problema v veliki meri vpliva na prepustnost celotnega sistema in je torej ena od najvažnejših odločitev pri snovanju mikroročunalniškega sistema. Hitrost je tudi vzrok, da ne uporabimo programske temveč aparaturne rešitve. Sistem za dodeljevanje vodila po vnaprej določeni shemi se imenuje arbitracija.

Druge pomembne odločitve pri snovanju protokola vodila so še

- sinhrono ali asinhrono vodilo: odločiti se moramo, ali bodo vse enote v računalniku delale z istim taktom, ki se prenaša po skupnem vodilu (sinhrono) ali bodo lahko njihove hitrosti različne (asinhrono). Sinhroni način je bistveno enostavnejši, asinhroni pa omogoča kasneje vključevanje novejših, hitrejših enot.
- multipleksirano vodilo ali ne: ker zaradi same zasnovave mikroprocesor in resursi ne potrebujejo istočasno na linijah naslova in podatkov, lahko po istih linijah pošljemo naprej adresu, nato pa vsebino teh celic. Za-

radi manj linij je vodilo manjše, uporabi se manj vezij, vendar ne dosegamo takšnih hitrosti kot pri nemultipleksiranih vodilih.

Članek teh problemov ne obravnava. Dotaknili se bomo le tematike arbitraže.

SPLOŠNO O ARBITRACIJSKIH ALGORITMIH

Glavni nalogi arbitracijskega sistema sta preprečiti istočasno dostop dveh ali več procesorjev do vodila in dodeljevanje vodila procesorjem po vnaprej predpisanem optimalnem pravilu.

Osnovne odločitve o tipu arbitracijskega algoritma so vezane na naravo naloge, ki bi jo naj opravljal. Če gre za sprotni sistem, nas bo zanimal prioritetni (ali vsaj delno prioritetni) algoritem, če želimo poslovni sistem z več procesorji in pametnim razdeljevanjem nalog, bo nam bolj ustrezal FIFO ali Round Robin algoritem (rotirajoči prioritetni algoritem).

- Kot za pridobitev nas zanimajo tudi algoritmi za sproščanje vodila. Poznamo dva glavna principa
- sprostitev, ko je prenos končan (release when done, RWD) in
 - sprostitev na zahtevo (release on request - ROR).

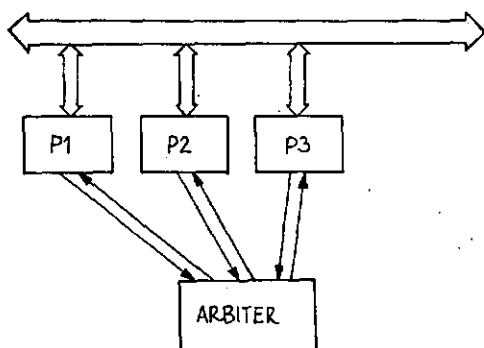
Pri prvem enota, ki je zmagala v arbitracijskem postopku, sprosti vodilo, ko je izvedla prenos podatkov, ne glede na to, katera enota je medtem zahtevala vodilo. Pri drugem pa enota med prenosom podatkov spremlja linijo, preko katere ji arbiter signalizira, da mora prekiniti s

komunikacijo, ker je vodilo zahtevala po prioriteti višja enota.

Iz samega delovanja vodila lahko sklepamo, da poleg vodila in čistih modulov obstaja še sistem, ki skrbi za arbitracijo. Ta arbitracijski sistem je lahko samostojni modul, lahko pa je porazdeljen na vse potencialne uporabnike vodila. Glede na način povezave modulov nanj ločimo nekaj osnovnih tipov:

- Zvezdasta povezava:

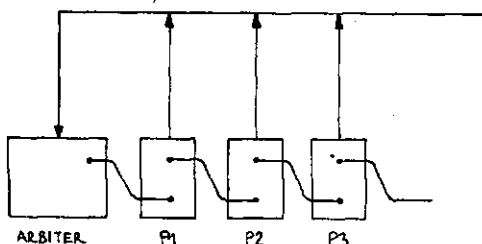
Vse naprave, ki lahko zahtevajo vodilo, so povezane na arbiter mimo vodila s po dvema posebnima linijama. Po eni enota zahteva vodilo (Bus Request), po drugi dobi odobritev (Bus Grant). Protokol je izredno enostaven, arbiter lahko dela po kakršnikoli metodi. Slabe strani so drago posebno ožičenje, na vodilu ni informacije o tem, kdo trenutno zaseda, na arbiter ni mogoče programsko vplivati.



Slika 1: Zvezdasta povezava na arbiter

- Daisy chain:

Na vsakem konektorju je par priključkov za arbitražo. Signal s prejšnjega konektorja gre na en priključek, z drugega pa na naslednji konektor. Na ta način je reali-



Slika 2: Daisy chain

zirana zaporedna povezava modulov.

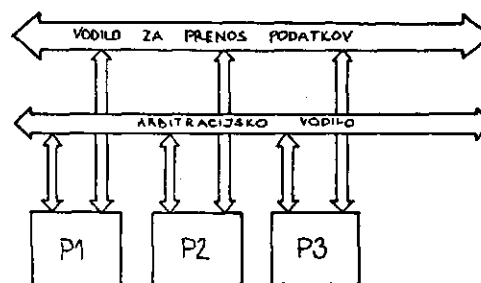
Kadar hoče naprava vodilo, pošlje po "wire - OR" liniji signal arbitru in čaka na odobritev na vhodnem priključku. Vsak modul ima vhodni in izhodni priključek kratko staknjen, razen če tudi sam želi vodilo. Kadar torej dve ali več naprav istočasno pošljeta zahtevo po vodilu, ga bo dobila tista, katere konektor je fizično bližje arbitru.

Slabosti tega sistema so:

- počasnost, saj vsaka plošča doda nekaj zakasnitve, dokler se signal ne vrne do enote, ki je zahtevala vodilo
- v vsakem konektorju mora biti plošča ali slepi konektor s prevezo, ki omogoči prenos signala
- o arbitraciji je na liniji še vedno malo podatkov.

- Shema z naslovno - prioritetnimi linijami

To posebno shemo uporablja več sodobnejših vodil in tudi standard IEEE P 896 Futurebus. Na vodilu je 4 do 8 arbitracijskih linij, na katere skušajo posamezni moduli vpisovati svoje prioritete številke po posebnem algoritmu, opisanem v 4. poglavju. Čigar številka je ostala na vodilu, je dobil odobritev za prenos podatkov. Razširitev osnovnega algoritma, da se moduli po principu "poštenosti" sami odpovedujejo vodilu in s tem omogočajo, da tudi moduli z nižjimi prioritetskimi številkami dobijo dostop do vodila, bistveno izboljšanje lastnosti, povezane s sodobnimi izsledki o primernosti posameznih arbitracijskih algoritmov. S tem se namreč približujemo enakopravnosti modulov. Prioritete ostanejo le za reševanje istočasnih zahtev vodila.



Slika 3: Naslovno prioriteta arbitraža

ARBITRACIJSKI ALGORITEM PRI VME VODILU

Eno najbolj znanih vodil je gotovo VME vodilo. Njegov razvoj sega v pozna sedemdeseta leta, ko je pri Motoroli skupina inženirjev razvijala Exormacs na osnovi

MC 68000 procesorja. Njegovo vodilo se je imenovalo VERSAbus. Kmalu pa so v evropskih podružnicah Motorola ugotovili, da se VERSAbus ne bo obnesel kot standard, zato so ga prenesli na evropski format tiskanega vezja in ga imenovali VME. Leta 1983 je IEEE začela postopek za standardizacijo VME vodila ter mu dala šifro P1014. Sprejetje standarda se pričakuje konec leta 1985. Istočasno bi ga naj sprejela tudi IEC.

VME sistem je sestavljen iz štirih skupin signalnih linij (vodil) in zbirke funkcionalnih modulov, ki predstavljajo vmesnike med napravami in vodili.

Štiri vodila, ki sestavljajo VME vodilo, so vodilo za prenos podatkov (DTB - Data Transfer Bus), arbitracijsko vodilo, prioriteto prekinitveno ter vodilo za notranje potrebe (ura, inicializacija, napake ipd...).

Vodilo je bilo prvotno namenjeno enoprocorskim sistemom, v katerih so določene enote lahko zahtevale prenos podatkov (npr. DMA kontrolerji, disk kontrolerji itd...). Z razvojem večprocesorskih sistemov pa se VME vodilo uporablja tudi zanje.

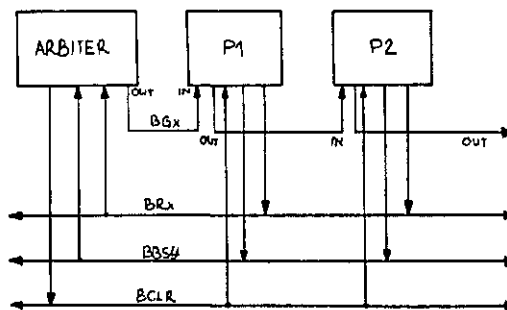
Arbiter na VME vodilu je lahko izveden kot ena od treh opcij:

- PRI (prioritetni algoritem): arbiter dodeli vodilo po stalnih prioritetah modulov, ki dajejo zahteve na linije, od najvišje (BR3) do najnižje (BR0)
- RRS (Round Robin Select): arbiter dodeljuje vodilo po principu rotirajočih prioritet. Če je imel vodilo modul na nivoju N, bo imel najvišjo prioriteto modul na nivoju N-1, dokler ne pridemo do začetka (N = 0). Tedaj ima najvišjo prioriteto spet N = 3.
- ONE (edini nivo): arbiter upošteva samo linijo BR3 in obravnava Daisy Chain strukturo na tem nivoju.

Arbiter je vedno v fizično prvem konektorju na vodilu. To je povezano s principom Daisy Chain. Lahko je samostojna plošča ali pa le razširitev univerzalne procesorske plošče, ki izpolnjuje zahteve minimalnega arbitra.

Razen pri opciji ONE arbiter sprejema zahteve po vodilu na štirih linijah BR0 do BR3, ki so izvedene v tehniki z odprtím kolektorjem, tako, da lahko povežemo izhode več modulov. Vsaka linija ima odgovarjajočo linijo za odobritev vodila (BGx-grant), izvedeno kot Daisy Chain. Če je vodilo prosto, ko je arbiter dobil zahtevo, takoj pošlje grant po ustrezni liniji. Signal potuje med moduli, ki imajo vhodne in izhodne linije sklenjene, dokler ne naleti

na modul, ki je nanj čakal. Ta dobi vodilo. Ko ga je nehal uporabljati, ga sprosti in pošlje grant za naslednjo zahtevo. Informacijo o zasedenosti vodila nosi linija BBS4 (Bus Busy). Arbiter lahko pri prioritetenem načinu tudi zahteva takojšnjo sprostitev vodila po liniji BCLR (Bus Clear), kar odgovarja načinu sproščanja na zahtevo (ROR).



Slika 4: Shematski prikaz delovanja arbitracijske sheme na VME vodilu za eno Daisy Chain linijo

Za arbitražo lahko uporabimo prioriteten ali krožni (Round Robin) algoritem. Na vsakem od nivojev se ustvari Daisy Chain struktura, pri čemer ima modul, ki je bližje arbitru, višjo prioriteto, oziroma bo večkrat dobil odobritev za uporabo vodila. Novejše raziskave kažejo, da prioriteten način ni najoptimalnejši pri dodeljevanju resursov. To ne velja za sprotne sisteme, kjer zaradi narave taskov mora obstajati prioriteta lestvica. Največja prepustnost sistemov se doseže pri enakovrednosti procesorjev in ustrezni razdelitvi nalog med njimi. V ta namen bi bil najustreznejši arbitracijski algoritem čakalna vrsta po principu FIFO. Te pa s konceptom, kot ga ima VME vodilo, ni mogoče uresničiti.

Sam problem nastopi pri tesni povezanosti procesorja v sistemu. Če dodamo vsakemu procesorju svoj del lokalnega pomnilnika in tako samostojne naloge, da ne bo pogosto zahteval povezave z drugimi moduli, bo vodilo razmeroma nezadostno in se bo vrsta le redko zgradila. Tedaj strežni algoritem ne bo imel prehudega vpliva na prepustnost sistema. Tipični primer takšne arhitekture so računalniški sistemi za vodenje robotov.

Pogosta je zasnova, kjer vsako os robota vodi svoj računalnik, vsi pa so povezani preko VME vodila s skupnimi globalnimi resursi - globalni pomnilnik, disk in druga periferija. Vsak računalnik je na nivoju premikov svoje osi avtonomen, ima svoj lokalni pomnilnik in direktno upravlja preko vmesnikov z izvršilnimi organi. Skloplje-

nost sistema je v tem primeru še dovolj lahka, da slabe strani prioritete sheme ne pridejo usodno do izraza.

SPLOŠNEJŠI ARBITRACIJSKI ALGORITEM

Ker so večprocesorski sistemi vse številnejši, je nastala potreba po novem vodilu, namenjenem prav zanje. V ta namen so pri IEEE ustanovili posebno komisijo, ki razvija novo standardno vodilo IEEE P896.

Osnovna ideja o arbitraciji datira v leto 1966. Kasneje je bila predelana in prirejena za sodobne zahteve ter uporabljena v standardih IEEE 696, P896 ter v NuBus (Texas Instruments) in Multibus II (Intel).

Vsaka enota ima svojo N-bitno arbitracijsko številko, vodilo pa N-bitno arbitracijsko vodilo na principu odprtega kolektorja. Ko dobi enota, ki je zahtevala vodilo, signal za začetek arbitracije, skuša na arbitracijske linije vpisati svojo številko. Če je na vodilu že višja prioriteta, kot je njena, umakne manj pomembne bite. Enoti, katere prioriteta ostane na vodilu ob koncu arbitracije, se dovoli prenos podatkov. Predno ta sprost vodilo, sproži ponovno arbitracijo, ki omogoči naslednji najpomembnejši enoti dostop do vodila.

Ker smo s tem pravilom še vedno pri prioritetah mu dodamo še pravilo "poštenja", ki prepoveduje enoti, ki nima izrecne časovne omejitve, zahtevati vodilo, ki ga je sprostila, dokler ga ne dobijo vsi drugi prosilci. Na ta način dobimo gotovo najpopolnejšo vodilo, vendar z najbolj zapletenim arbitracijskim vezjem, ki mora biti na vsakem modulu v sistemu.

Na tem mestu je treba razčistiti s pojmom prioritete na vodilu. Namen prioritete v zvezi s pravilom "poštenosti" je rešiti problem istočasnih zahtev. V zgoraj opisanem sistemu ima prioriteta malo skupnega z odločanjem, katera enota bo največkrat dobila vodilo in nima nič skupnega s prioriteto taskov. Taski z nizko prioriteto imajo mnogokrat na vodilu visoko prioriteto (masovni prenos podatkov itd...).

ZAKLJUČEK:

VME vodilo je bilo koncipirano za enoprocorske sisteme. Sam koncept sicer podpira tudi večprocesorsko arhitekturo, vendar ne moremo doseči optimalne propustnosti, kot jo teoretično lahko določimo. VME sistem pa ima druge prednosti. Ker je kot standard izredno široko razširjen in priznan, obstaja na tržišču mnogo modulov, ki jih lahko uporabimo. Od zahtevnosti problema, ki bi ga radi rešili z našim računalnikom bo odvisno, ali se bomo zadovoljili z VME ali bomo iskali optimalnejše arbitracijske algoritme.

LITERATURA

- /1/ Rozman, I., Colnarič, M.: Modeliranje MP/MC arhitektur s skupnim vodilom, Jugoslovanski mednarodni simpozij za računalniško tehnologijo in probleme informatike, Informatica 83, Ljubljana 1983, str. 14-18
- /2/ Colnarič, M., Rozman, I.; Simulacija multiprocorskega računalnika s skupnim vodilom, Jugoslovnsko svetovanje o mikroprocesorskim sistemima MIPRO 83, Rijeka 1983, str. 2.32 - 2.38
- /3/ Rozman, I., Colnarič, M., Bonačič, D.: Analiza učinkovitosti arhitekture s skupnim vodilom pri statistično neenakem zasedanju vodila, Jugoslovnsko svetovanje o mikroprocesorskim sistemima MIPRO 84, Rijeka 1984, str. 3.80 - 3.84
- /4/ VMEbus Manufacturers Group: VMEbus Specification Manual, Rev. B, August 1982
- /5/ Gustavson, D.B.: Computer Busses - A Tutorial, IEEE MICRO, August 1984, str. 7 - 22
- /6/ Taub, D. M.: Arbitration and Control Aquisition in the Proposed IEEE 896 Futurebus, IEEE MICRO, August 1984, str. 28 - 41
- /7/ Fischer, W.: IEEE P1014 - A Standard for the High-Performance VME Bus, IEEE MICRO, February 1985, str. 31 - 41