

## Simultaneous multithreading (Hyperthreading)

Superskalarni procesori, zahvaljujući ugrađenoj dataflow mašini koja po dataflow principu obrađuje dinamički graf zavisnosti po podacima velikom brzinom, postali su dominantni procesori opšte namene. Međutim, kada su urađene analize stvarne iskorišćenosti ugrađene dataflow mašine, pokazalo se da ta iskorišćenost iznosi 30-40%. Razloga za to je puno, a osnovni su sledeći: Keš promašaj za data ili instrukcijski keš, Load iz L2 ili L3 keša, greške u predikciji grananja, ...

Nametnula se ideja da se efikasnije iskoriste resursi procesora tako što bi se procesoru dao drugi posao, ali bez izmena konteksta, jer bi to dugo trajalo. Osim toga, razmatrane su ideje da se u dataflow mašinu nekako ubaci više niti izvršavanja uz minimalno povećanje resursa mašine. Motiv za takvo razmišljanje je činjenica da, statistički posmatrano, graf sa  $N$  čvorova ima manji paralelizam od dva nepovezana grafa sa  $N/2$  čvorova i još manji paralelizam od četiri nepovezana grafa sa  $N/4$  čvorova. Kako niti imaju nepovezane grafove zavisnosti po podacima na nivou registarskih operacija, puštanjem više niti se dobija ukupan viši nivo paralelizma.

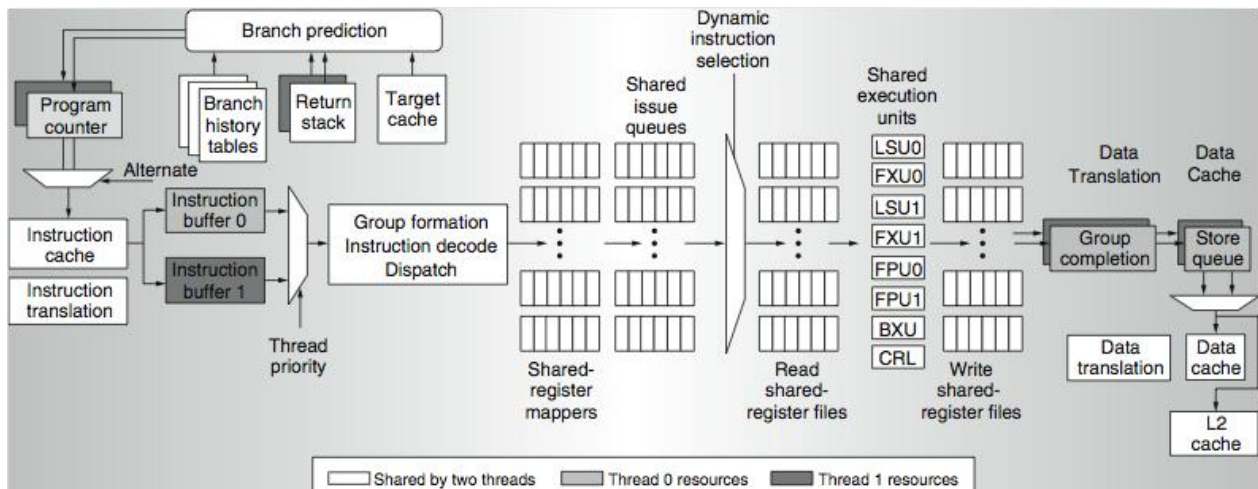
Osnovno pitanje je koliki je obim dodatnog hardvera potreban za svaku nit i da li je to isplativo. Jasno je da za svaku nit mora da se stvori iluzija da ima na raspolaganju kompletan procesor. Svaka nit zato mora da ima svoj bafer za dinamički trag, svoju predikciju grananja, svoje registre koji čuvaju in order stanje arhitekturno stanje niti ... Odvojena predikcija grananja ne znači nužno dupliranje hardvera, već se samo dupliraju globalne istorije niti, a resursi za sopstvene istorije, automati i sl. se podele između niti. Isto tako se može smanjiti dubina bufera za niti tako da se logika za dinamički trag malo menja.

Dalje će se razmatrati slučaj tag indexed mašine. Ona na ulazu u ugrađenu dataflow mašinu može da ima ekvivalent multipleksera koji odlučuje iz koje niti se trenutno ubacuju instrukcije u ugrađenu dataflow mašinu. Na takvom multipleksoru se može na osnovu željenog prioriteta niti definisati grubo procenat instrukcija iz svake od niti u ugrađenoj dataflow mašini. Osim toga, kada se u jednoj od niti dogodi npr. promašaj u kešu, tada multiplekser počne naglo da uzima samo instrukcije iz niti koja ne čeka i da tako podigne iskorišćenje dataflow mašine.

Osnovno pitanje je kako razdvojiti niti u jedinstvenoj dataflow mašini. To se postiže tako što obe niti imaju zajednički free tag FIFO. Kada je tag dodeljen nekoj niti, on ne može da se pojavi u drugoj niti do njegove dealokacije. Tako se stvaraju disjunktni skupovi tagova (fizičkih registara) za niti i time ostvaruju nepovezani grafovi simultano izvršavanja niti. Svaka nit mora da ima i sopstveni ROB, ali se i oni primarno realizuju deljenjem jedinstvenog ROB na dva manja uz dosta dodatne logike. Kao posledica, svaka nit mora da ima sopstveni rename file.

Koliko je niti optimalno nije moguće jednoznačno odrediti, zato što dodatna kompleksnost hardvera jako zavisi od načina implementacije komponenti superskalarnog procesora. Gruba procena je da porast broja tranzistora od 10-25% donosi oko 30-40% poboljšanja ukupnih performansi za slučaj dve niti. Sa povećavanjem broja niti se ne dobija mnogo, a značajno se usložnjava procesor, tako da su retki procesori koji imaju više od 4 niti po jezgru. Danas dominiraju procesori sa dve niti po jezgru. Na Sl. 1. je prikazan IBM Power 5 koji koristi simultaneous multithreading sa dve niti. Sa belom bojom su označeni zajednički delovi logike za obe niti, a sa nijansama sive zasebni delovi za niti. Među zasebnim delovima

nije označen deo prediktora grananja, jer se pretpostavljalo da nemaju globalnu istoriju dinamičkog traga niti. Boje ostalih delova slike su saglasne sa ostalim prethodnim objašnjenjima šta je zajedničko, a šta je zasebno za niti.



Sl. 1. IBM Power 5 koji koristi simultaneous multithreading sa dve niti. Naznačeni su deljeni resursi i zasebni resursi niti

Simultaneous multithreading donosi i neke probleme koji se teško mogu izbeći. Jedan je smanjena lokalnost u vremenu i memoriji, što dovodi do češćih promašaja keša. Druga je vezana za spekulativnost izvršavanja obe niti po kontroli, što može da dovede do situacija gde jedna nit, pod specijalnim okolnostima, može da čita podatke druge niti. To ne bi bilo moguće da nije kombinacije simultaneous multithreading-a i spekulativnosti.

Često se u kontekstu simultaneous multithreadinga govori o virtuelnim procesorima, jer niti nemaju načina da razlikuju da li raspolažu celokupnim procesorom ili ga dele sa još nekom ili nekim nitima. Današnja snažnija procesorska integrisana kola sa superskalarnim jezgrima imaju tipično 4-6 jezgara na jednom integrisanom kolu i nekada deljene L3 keš memorije. Jezgra dele memoriju kao multiprocesori sa zajedničkom memorijom i performanse celog memorijskog podsistema postaju ključne. Zato se na integrisanim kolima realizuju kompletni memorijski podsystemi interleaved memorije sa pipeline registrima. Trenutno aktuelne DDR 4 memorijske komponente imaju po 16 memorijskih banki u integrisanom kolu i donjih 4 bita adrese interno selektuje banke, pa se prenos blokova podataka iz DDR 4 memorije obavlja veoma velikom brzinom.