

AltiVec vs. Xlf

Mikko O

23. syyskuuta 2004

Alla on taulukoitu erilaisten vektorilaskutoimitusten nopeudet ja nopeuksien suhde, kun ne on tehty normaalia Fortrania ja AltiVec-koodia käyttämällä.

command	$T_F(\mu s)$	$T_{AV}(\mu s)$	T_F/T_{AV}
sum(vec1)	2.21	3.85	0.57
1/vec1	34.48	3.96	8.71
sqrt(vec1)	125.35	6.59	19.02
vec1+vec2	8.04	3.92	2.05
vec1*vec2	8.73	3.16	2.76
vec1/vec2	34.22	6.34	5.40

Selityksiä:

- `vec1`, `vec2`: 2048-paikkaisia *single precision* vektoreita
- `sum(vec1)`: vektorin `vec1` komponenttien summa
- `1/vec1`: vektorin komponenttien käänteisluvut
- `sqrt(vec1)`: vektorin komponenttien neliöjuuret
- `vec1+vec2`: kahden vektorin summa
- `vec1*vec2`: kahden vektorin kertolasku (komponenteittain)
- `vec1/vec2`: kahden vektorin jakolasku (komponenteittain)

Huomioita:

- Fortran-koodi käännettiin optiolla `-O5`. Jonkinlaista nopeutumista voisi saada aikaan, jos oikein yrittäisi, mutta käytössä oli Xlf:n beta-versio, jossa kaikki optimointioptiot eivät täysin toimi.
- C:llä kirjoitettu AltiVec-koodi käännettiin `gcc`:llä optioilla `-fast -mcpu=7450`. Parempaa en tiedä...
- Vektorin komponenttien summan laskemisessa Fortran oli nopeampi, mutta toisaalta tuollaisen summan laskeminen AltiVecilla ei ole järkevää.
- Tämä testi tehtiin OS X:llä. Pitää vielä kokeilla samaa PPCLinux-puolella.
- Pitäisi vielä tehdä samat testit C:llä ja katsoa, miten se pärjää Fortranille ja AltiVecille.