

## Coarse Grain Parallelization and Speculative Execution

Toshihiro Ozawa<sup>1,2</sup>, and Akira Asato<sup>1,2</sup> (<sup>1</sup> APC Technology Group <sup>2</sup>Fujitsu Limited)



## **Example of Speculative Execution(1)**



## Example of Speculative Execution(2)



OMP SECTION CALL OMP SETLOCK(PI) CALL OMP\_UNST\_LOCK(PI) DO 0810 = 1, N373 DO 0813 = 1, N DO 0813 = 1, N1 DO 0813 = 1, N2 DO 0813 = 1, N2 DO 0813 = 1, N2 B MIN = TMP + AUI3XX+1/J AU33X53 CALL OMP\_SET\_LOCK(P) PF7 : EQ (J 0074 3 PF7 : EQ (J 0074 3) RESULT = TMP S0 CONTINUE

## Case Study of Procedure level Parallelization (CFP2000/168.wupwise)

ALPHA = ZDOTC(SIZE, RD, 1, R, 1) DO 100 ITER = 1, NITER CALL MATMUL(U, N1, N2, N3, N4, KAPPA, P, AUX, UD) BETA = ZDOTC(SIZE, RD, 1, UD, 1)	X:Input array <u>Y</u> :Output arra
DELTA = ALPHA/BETA CALL ZCOPY(SIZE, R, 1, S, 1) CALL ZAXPY(SIZE, -DELTA, UD, 1, S, 1)	
CALL MATMUL(U,N1,N2,N3,N4,KAFPAS, $AUX,T$ ) ST = ZDOTQ(SIZE,T,1,S,1) TT = ZDOTQ(SIZE,T,1,T,1)	
OMEGA = ST/TT CALL ZCOPY(SIZE, S. I, <b>R</b> , 1) CALL ZAXPY(SIZE, OMEGA, T, 1 <b>R</b> , 1)	
CALL ZAXPY(SIZE, DELTA, P. 1, X, 1) CALL ZAXPY(SIZE, OMEGA, S, 1, X, 1)	

DO 00 M 4 = 1, N2 D0 00 K 4 = 1, N2 Execution Committeent B8d         D1 0 0 0 0 0         D1 0 0 0 0         D1 0 0 0         D1 0 0         D1 0         D1 0 <th>DOUR EZ - 1, N2 002 RE12RO - P47-K2-R(E.2.K2)-A(E.1.2,R2) IF68(NN2N2, EQ.0) THEN P3-1 CALL OMP_UNSET_LOCK(P2) ESE P3-1 CALL OMP_UNSET_LOCK(P2) ENDIF Alpha Server GS160 Model 6/73 :Alpha 21264 (731 MHz) × 8 Alpha Digital Fortran Compiler: -v - arch ev6-05 11</th> <th>ALM = ZDOTOSIZE_ROL_R.) RIO = ALPHALPHA PEN PELTACOMEGA ALMA = ALPH CALL ZACVISZE_ROL_R.) CALL ZACVISZE_ROL_R.) CALL ZACVISZE_ROL_R.(J.) RNRM2 = DZNRMA_SIZE_R.) EVS = RNNM2 NRNM2 COS = PRNM2 NRNM2 COS = PRNM2 NRNM2 SOUTHER EVS 92 = FORMATREDISS) 12 10 CONTINUE</th>	DOUR EZ - 1, N2 002 RE12RO - P47-K2-R(E.2.K2)-A(E.1.2,R2) IF68(NN2N2, EQ.0) THEN P3-1 CALL OMP_UNSET_LOCK(P2) ESE P3-1 CALL OMP_UNSET_LOCK(P2) ENDIF Alpha Server GS160 Model 6/73 :Alpha 21264 (731 MHz) × 8 Alpha Digital Fortran Compiler: -v - arch ev6-05 11	ALM = ZDOTOSIZE_ROL_R.) RIO = ALPHALPHA PEN PELTACOMEGA ALMA = ALPH CALL ZACVISZE_ROL_R.) CALL ZACVISZE_ROL_R.) CALL ZACVISZE_ROL_R.(J.) RNRM2 = DZNRMA_SIZE_R.) EVS = RNNM2 NRNM2 COS = PRNM2 NRNM2 COS = PRNM2 NRNM2 SOUTHER EVS 92 = FORMATREDISS) 12 10 CONTINUE
Case Study of Procedure level Parallelization (CFP2000/168.wupwise)	Case Study of Procedure level Parallelization (CFP2000/168.wupwise)	Summary
•6 parallel tasks are extracted from the main loop * 2_1, 6_1 13 CALL MATMUL (LIN1, N2, N3, N4, KAPPA, P. ALK, UD) 14 BETA = ZDOT(SIZE, BD, 1, UD, 1) * 4_1, 5_1 * 1_2, 1_5 * 1_2, 1	horizontal sequential: -v -arch ev6 -O5 -fkapargs='.conc -ur=1' Procedure Parallel: -v -arch ev6 -O5 -fwapargs='.conc -ur=1	<ul> <li>We have eveloped the parallelizing module which exploits both speculative and non-speculative execution at coarse grain level.</li> <li>Task: loop or procedure</li> <li>Speculation: control</li> <li>It is comfirmed that the performance is increased by the speculative execution, which is suitable for it.</li> <li>Some procedures are extracted as a parallel task from Spec2000/168.wupwise.</li> <li>The performance has increased by17% on Alpha Server GS160 Model 6/73.</li> </ul>