

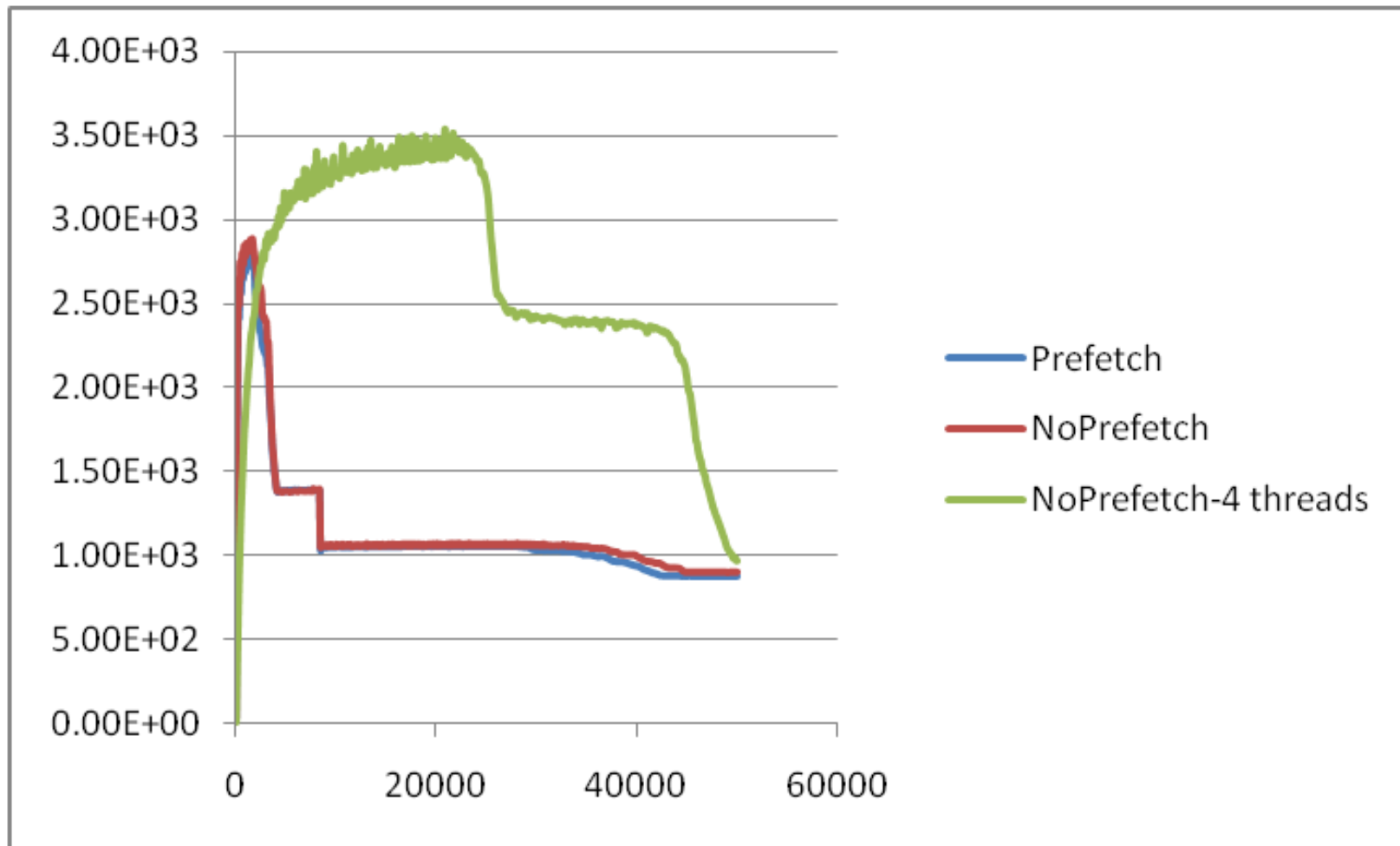
# Cache Optimization

# Memory Alignment Issues

- Cache Boundaries
- Page Boundaries
- Memory Banks

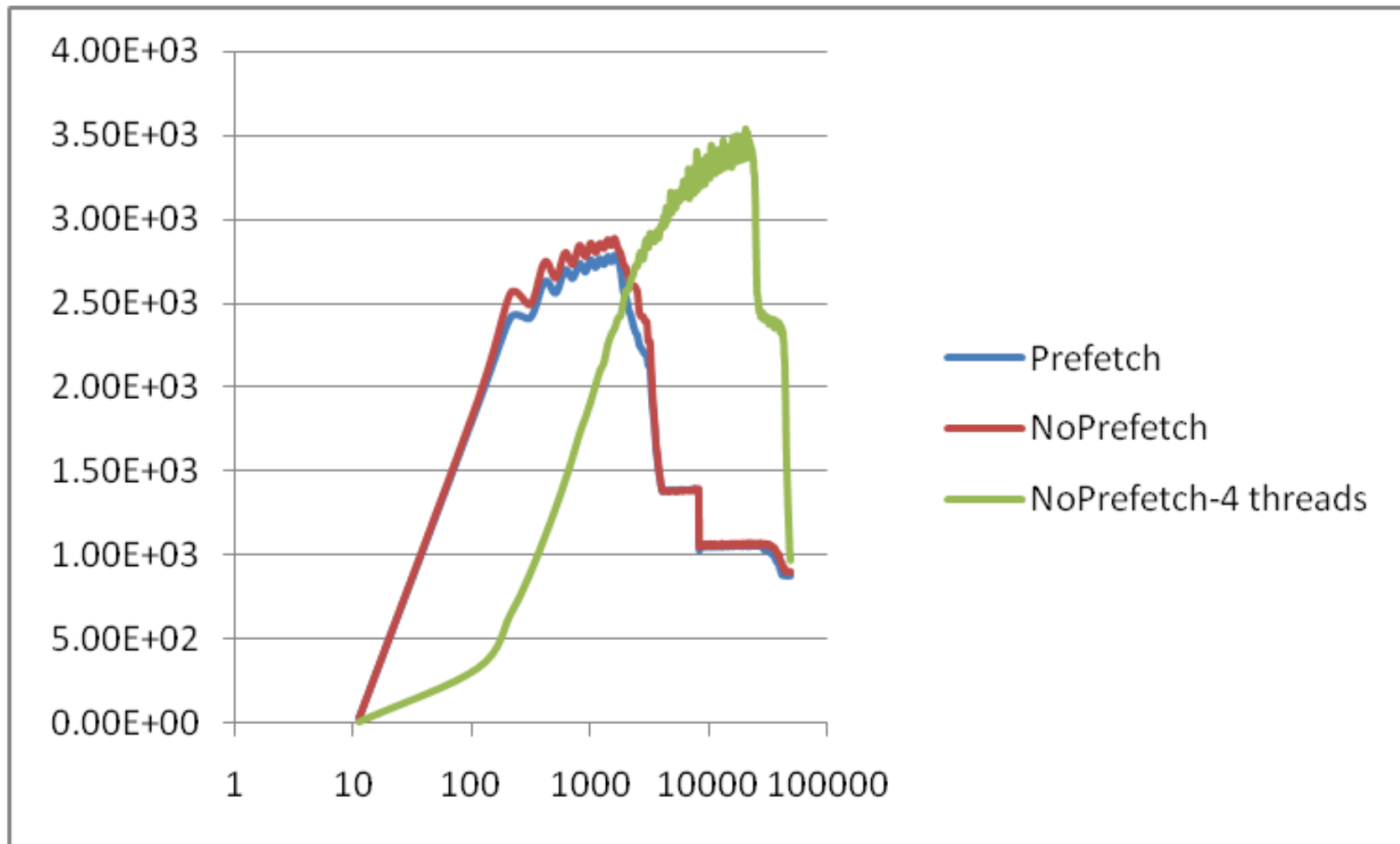
# Performance = F( Cache Utilization )

Stream Triad (MFLOPS)



# Performance = F( Cache Utilization )

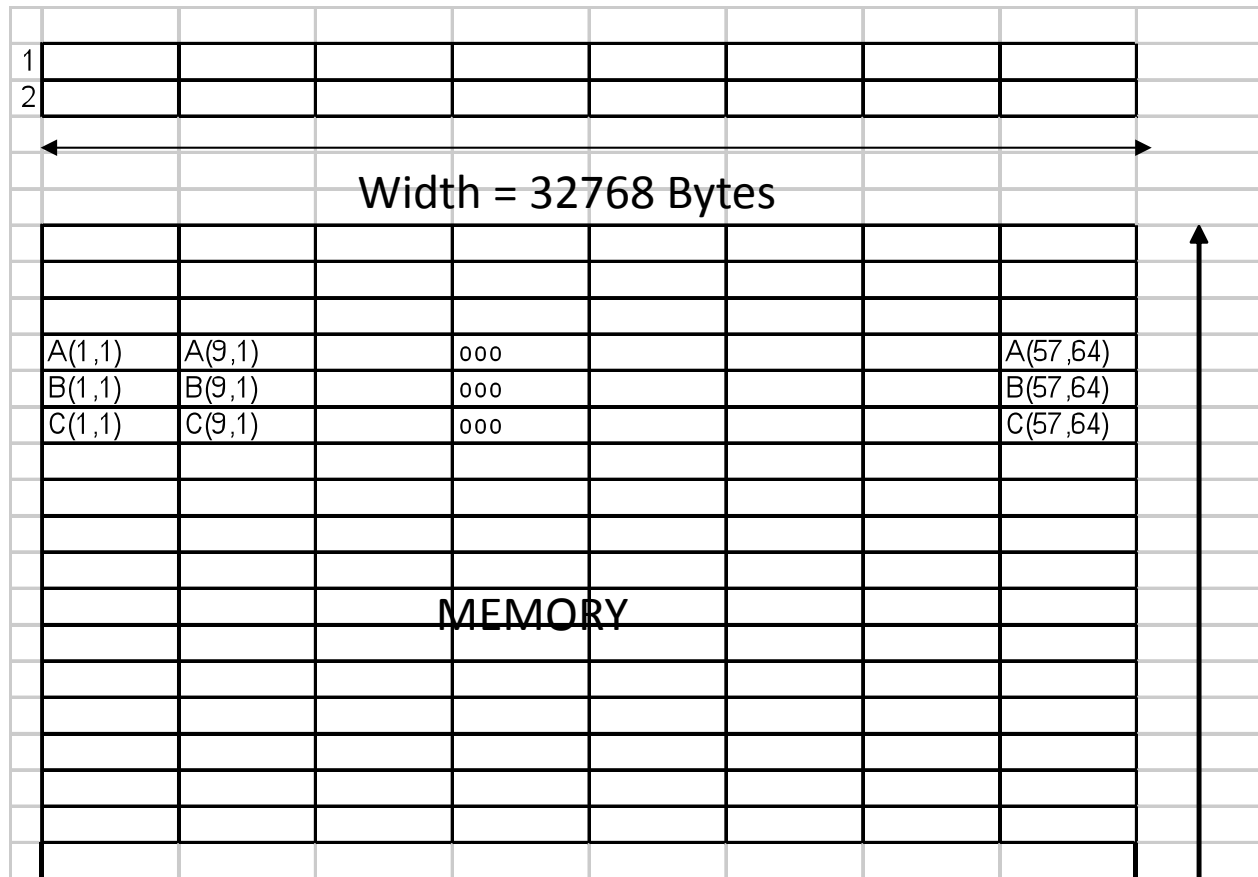
Stream Triad (MFLOPS)



<b>Real * 8</b>	<b>A(64,64),B(64,64),C(64,64)</b>				
<b>DO I = 1,N</b>					
<b>C(I,1) = A(I,1) +B(I,1)</b>					
<b>ENDDO</b>					

# Cache Visualization

Level 1 Cache



Level 1 Cache

65536 B

1024 Lines

8192 8B Ws

16384 4B Ws

2 way Assoc

Associativity Class

32768 B

512 Lines

4096 8B Ws

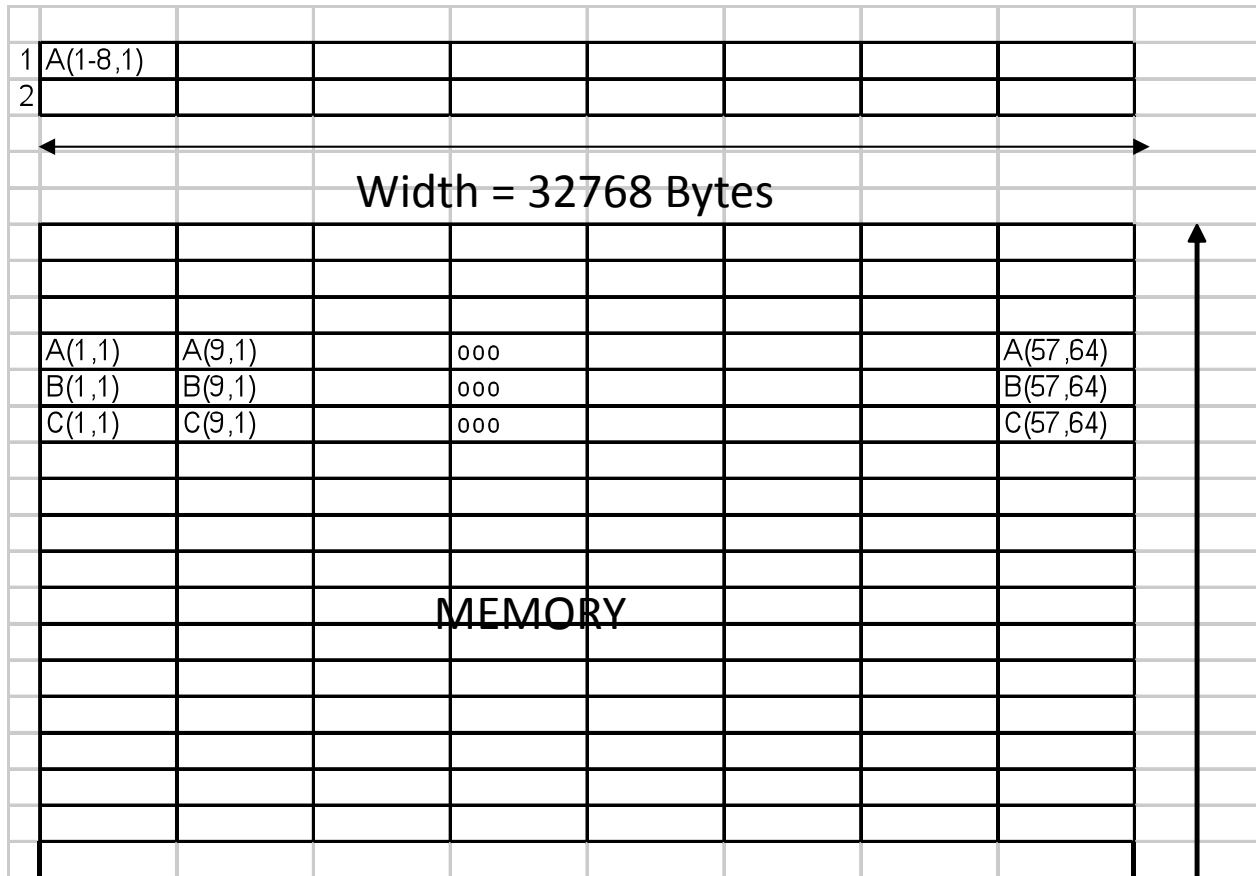
8192 4B Ws

$$64 * 64 * 8 = 32768 \text{ B}$$

# Consider the following example

<b>Real * 8</b>	<b>A(64,64),B(64,64),C(64,64)</b>			
<b>DO I = 1,N</b>				
<b>C(I,1) = A(I,1) +B(I,1)</b>				
<b>ENDDO</b>				
<b>Fetch A(1,1)</b>		<b>Fetch from M Uses 1 Associativity Class</b>		

## Level 1 Cache



## Level 1 Cache

65536 B

1024 Lines

8192 8B Ws

16384 4B Ws

2 way Assoc

## Associativity Class

32768 B

512 Lines

4096 8B Ws

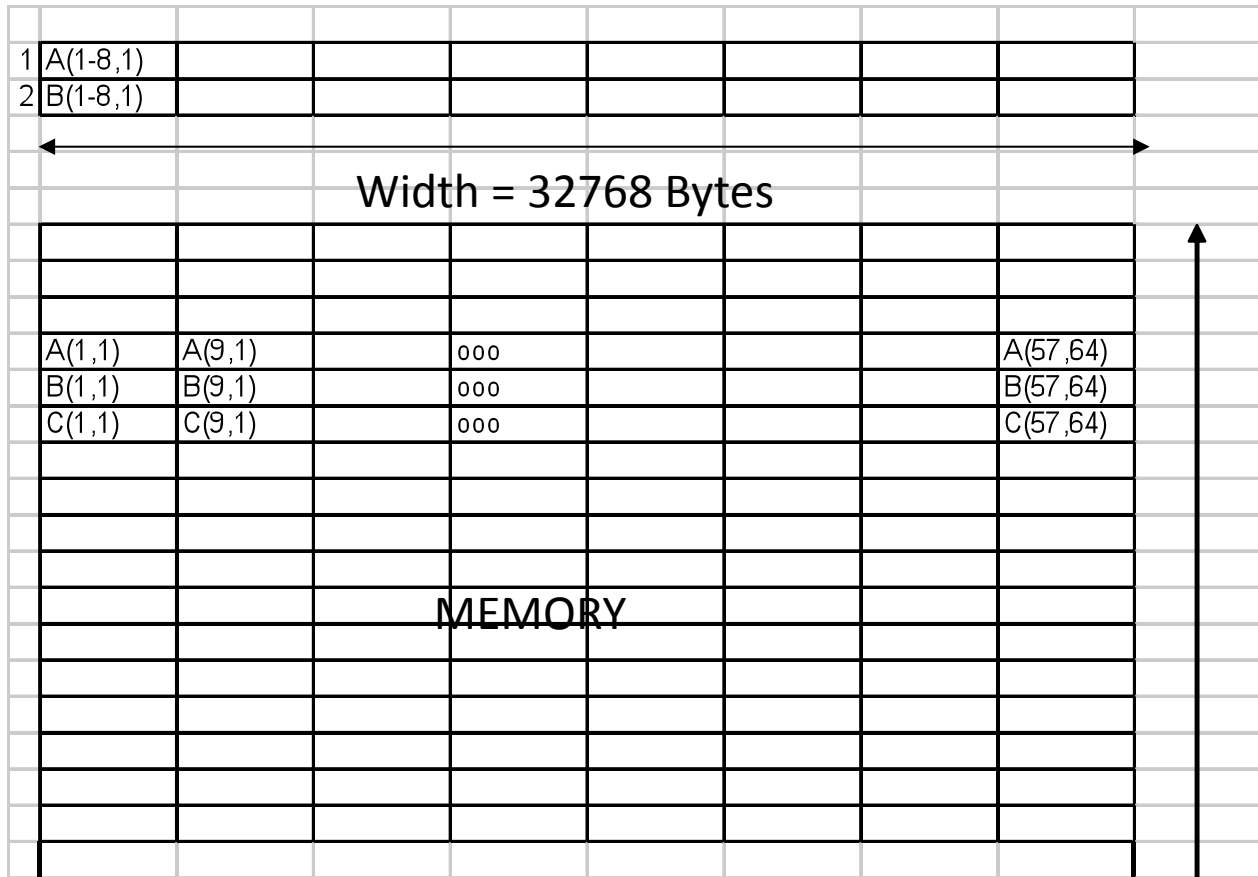
8192 4B Ws

$$64 * 64 * 8 = 32768 \text{ B}$$



<b>Real * 8</b>	<b>A(64,64),B(64,64),C(64,64)</b>			
<b>DO I = 1,N</b>				
<b>    C(I,1) = A(I,1) +B(I,1)</b>				
<b>ENDDO</b>				
<b>Fetch A(1,1)</b>		<b>Fetch from M Uses 1 Associativity Class</b>		
<b>Fetch B(1,1)</b>		<b>Fetch from M Uses 2 Associativity Class</b>		

## Level 1 Cache



## Level 1 Cache

65536 B

1024 Lines

8192 8B Ws

16384 4B Ws

2 way Assoc

## Associativity Class

32768 B

512 Lines

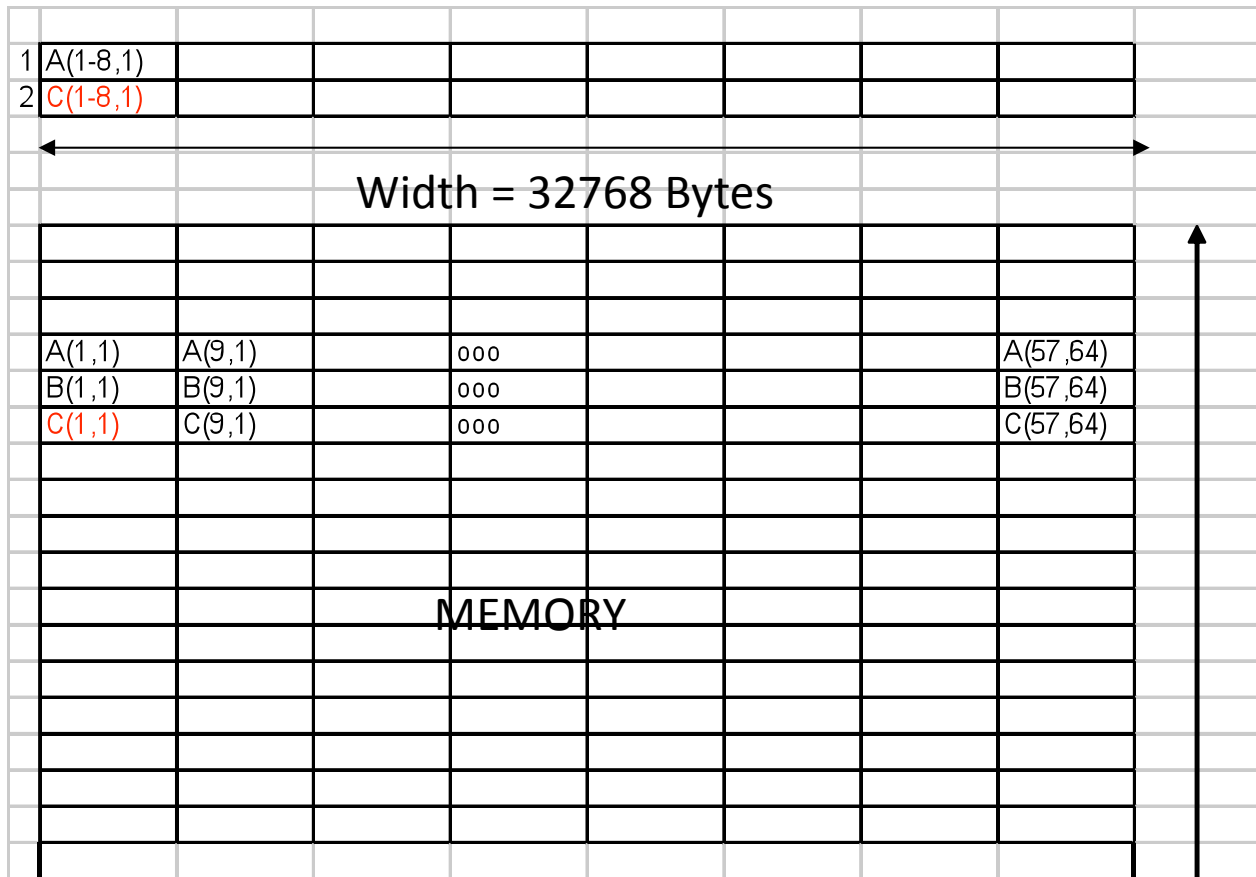
4096 8B Ws

8192 4B Ws

$$64 * 64 * 8 = 32768 \text{ B}$$

<b>Real * 8</b>	<b>A(64,64),B(64,64),C(64,64)</b>				
<b>DO I = 1,N</b>					
<b>C(I,1) = A(I,1) +B(I,1)</b>					
<b>ENDDO</b>					
<b>Fetch A(1,1)</b>		<b>Fetch from M</b>	<b>Uses 1 Associativity Class</b>		
<b>Fetch B(1,1)</b>		<b>Fetch from M</b>	<b>Uses 2 Associativity Class</b>		
<b>Add A(1,1) + B(1,1)</b>					
<b>Store C(1,1)</b>		<b>Fetch from M</b>	<b>Overwrites either 1 or 2 Associativity Class</b>		

## Level 1 Cache



## Level 1 Cache

65536 B

1024 Lines

8192 8B Ws

16384 4B Ws

2 way Assoc

## Associativity Class

32768 B

512 Lines

4096 8B Ws

8192 4B Ws

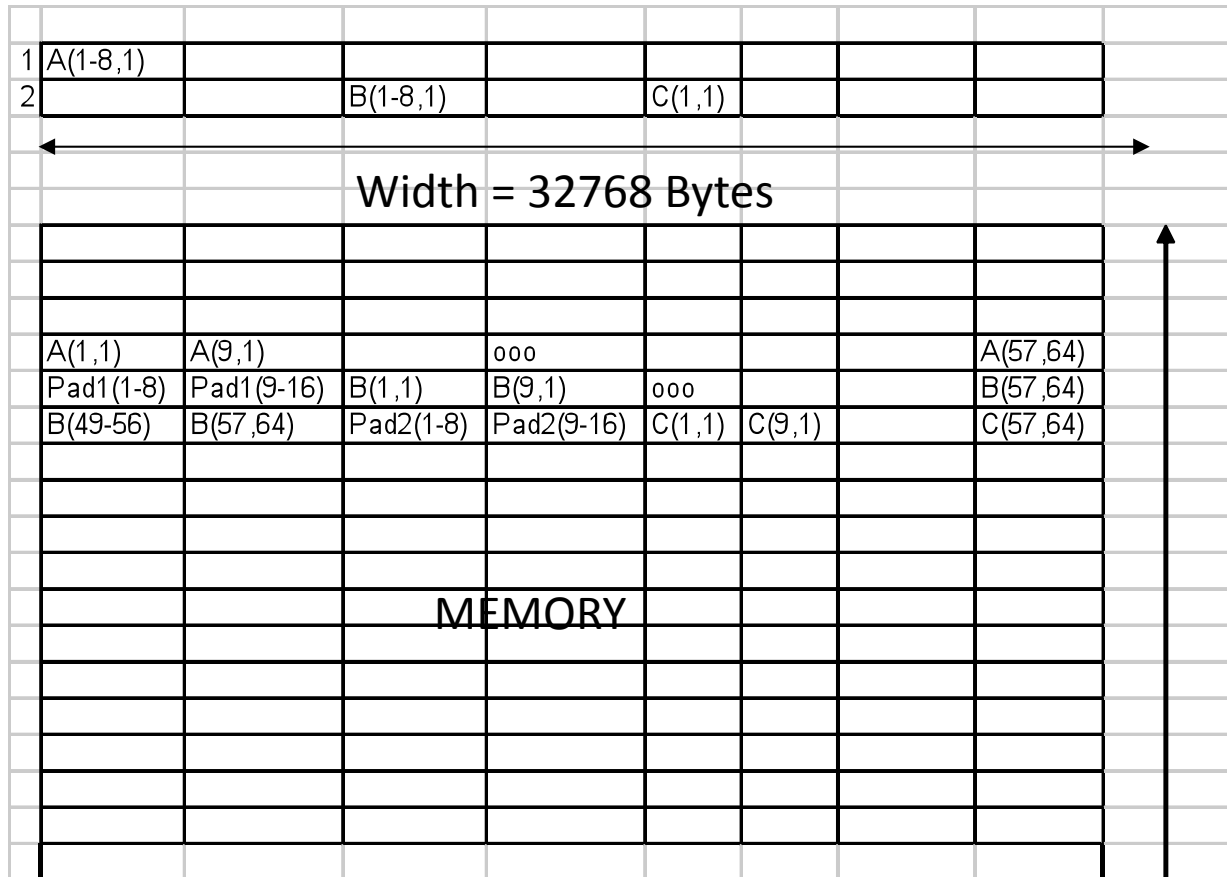
$$64 * 64 * 8 = 32768 \text{ B}$$

<b>Real * 8</b>	<b>A(64,64),B(64,64),C(64,64)</b>			
<b>DO I = 1,N</b>				
	<b>C(I,1) = A(I,1) +B(I,1)</b>			
<b>ENDDO</b>				
<b>Fetch A(1,1)</b>		<b>Fetch from M</b>	<b>Uses 1 Associativity Class</b>	
<b>Fetch B(1,1)</b>		<b>Fetch from M</b>	<b>Uses 2 Associativity Class</b>	
<b>Add A(1,1) + B(1,1)</b>				
<b>Store C(1,1)</b>		<b>Fetch from M</b>	<b>Overwrites either 1 or 2 Associativity Class</b>	
<b>Fetch A(2,1)</b>		<b>Fetch from L2</b>	<b>Overwrites either 1 or 2 Associativity Class</b>	
<b>Fetch B(2,1)</b>		<b>Fetch from L2</b>	<b>Overwrites either 1 or 2 Associativity Class</b>	
<b>Add A(2,1) + B(2,1)</b>				
<b>Store C(2,1)</b>		<b>Fetch from L2</b>	<b>Overwrites either 1 or 2 Associativity Class</b>	

# Must be a better Way

<b>Real * 8</b>	<b>A(64,64),pad1(16),B(64,64),pad2(16),C(64,64)</b>			
<b>DO I = 1,N</b>				
<b>C(I,1) = A(I,1) +B(I,1)</b>				
<b>ENDDO</b>				

## Level 1 Cache



## Level 1 Cache

65536 B

1024 Lines

8192 8B Ws

16384 4B Ws

2 way Assoc

## Associativity Class

32768 B

512 Lines

4096 8B Ws

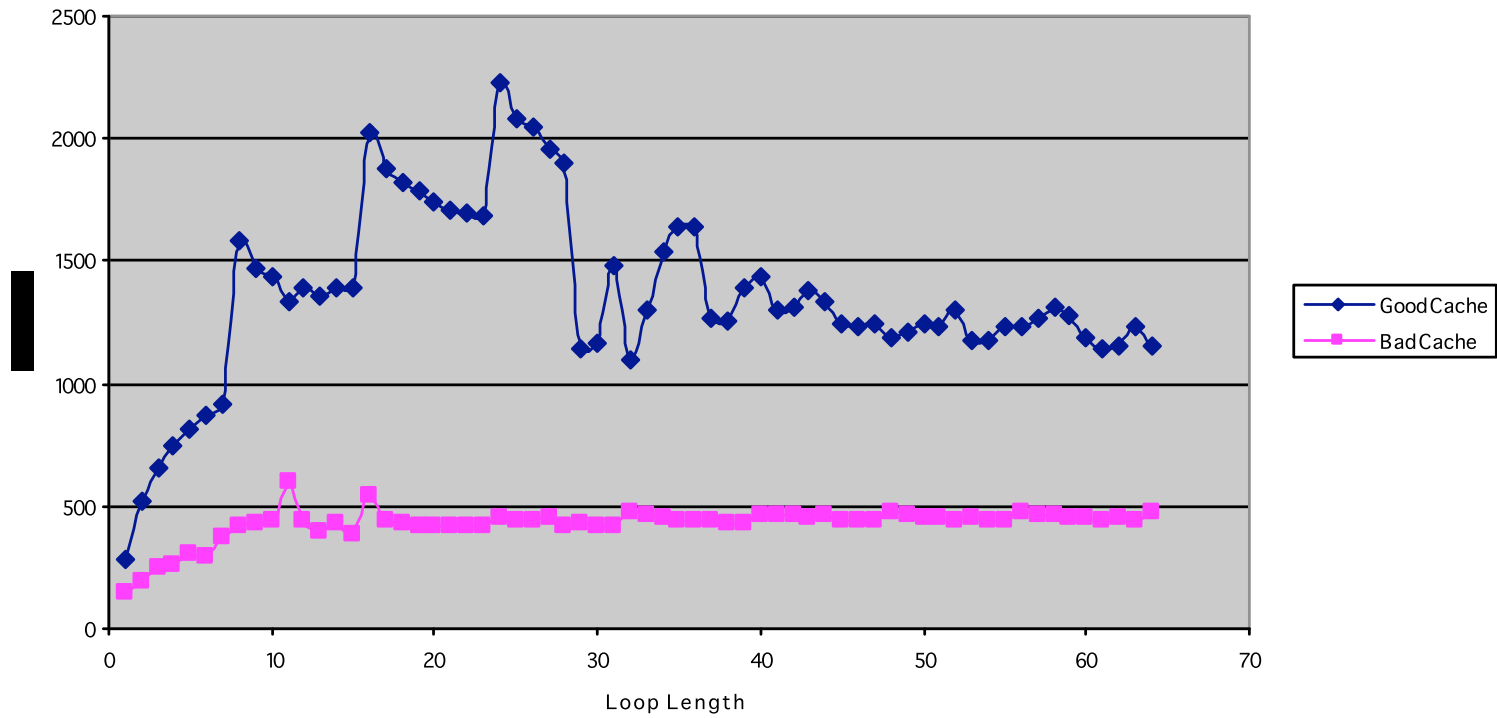
8192 4B Ws

$$64 * 64 * 8 = 32768 \text{ B}$$

<b>Real * 8</b>	<b>A(64,64),pad1(16),B(64,64),pad2(16),C(64,64)</b>			
<b>DO I = 1,N</b>				
	<b>C(I,1) = A(I,1) +B(I,1)</b>			
<b>ENDDO</b>				
<b>Fetch A(1)</b>		<b>Uses 1 Associativity Class</b>		
<b>Fetch B(1)</b>		<b>Uses 2 Associativity Class</b>		
<b>Add A(1) + B(1)</b>				
<b>Store C(1)</b>		<b>Uses 1 Associativity Class</b>		
<b>Fetch A(2)</b>		<b>Gets from L1 Cache</b>		
<b>Fetch B(2)</b>		<b>Gets from L1 Cache</b>		
<b>Add A(2) + B(2)</b>				
<b>Store C(2)</b>		<b>Gets from L1 Cache</b>		



### Cache Alignment Example



# Bad Cache Alignment

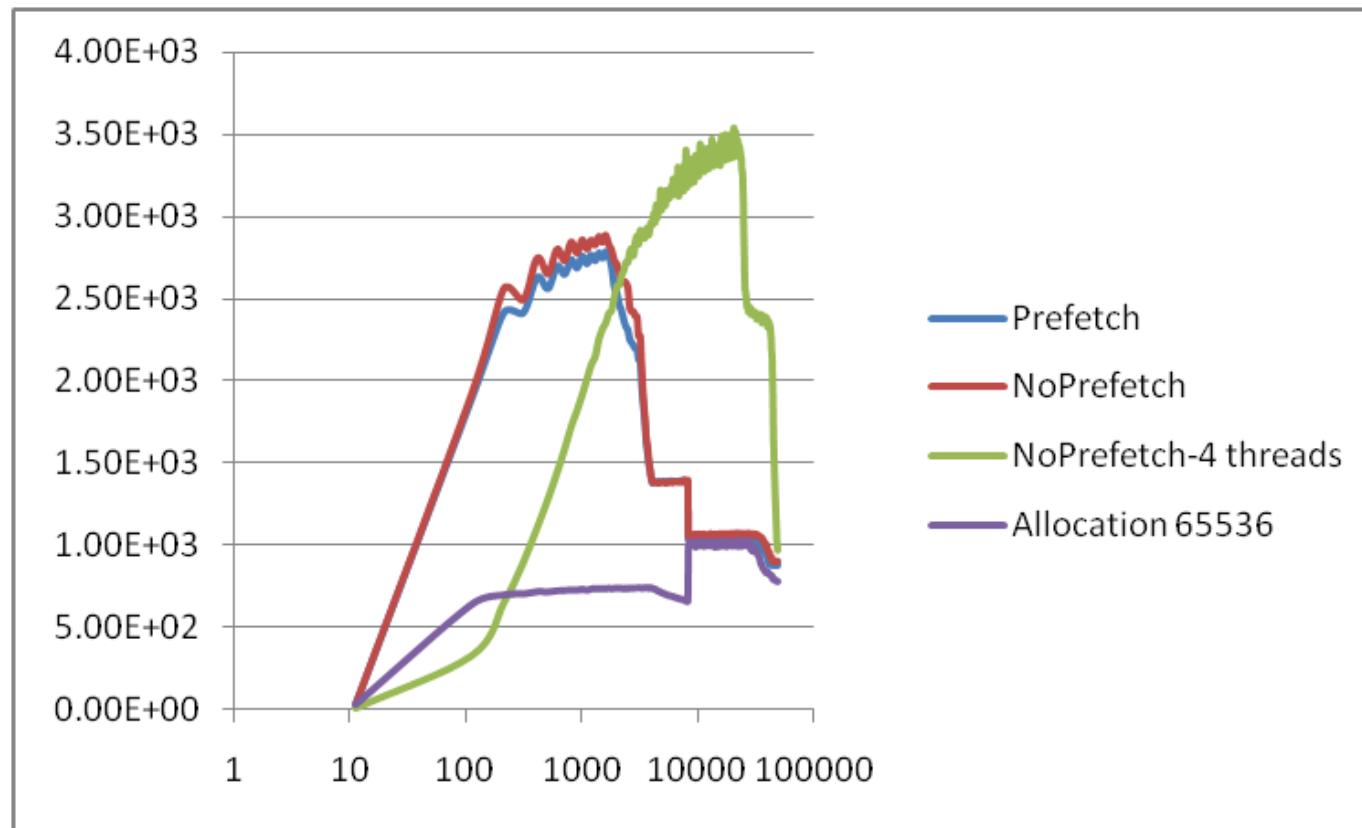
Time%		0.2%
Time		0.000003
Calls		1
PAPI_L1_DCA	455.433M/sec	1367 ops
DC_L2_REFILL_MOESI	49.641M/sec	149 ops
DC_SYS_REFILL_MOESI	0.666M/sec	2 ops
BU_L2_REQ_DC	74.628M/sec	224 req
User time	0.000 secs	7804 cycles
Utilization rate		97.9%
L1 Data cache misses	50.308M/sec	151 misses
LD & ST per D1 miss		9.05 ops/miss
D1 cache hit ratio		89.0%
LD & ST per D2 miss		683.50 ops/miss
D2 cache hit ratio		99.1%
L2 cache hit ratio		98.7%
Memory to D1 refill	0.666M/sec	2 lines
Memory to D1 bandwidth	40.669MB/sec	128 bytes
L2 to Dcache bandwidth	3029.859MB/sec	9536 bytes

# Good Cache Alignment

Time%		0.1%
Time		0.000002
Calls		1
PAPI_L1_DCA	689.986M/sec	1333 ops
DC_L2_REFILL_MOESI	33.645M/sec	65 ops
DC_SYS_REFILL_MOESI		0 ops
BU_L2_REQ_DC	34.163M/sec	66 req
User time	0.000 secs	5023 cycles
Utilization rate		95.1%
L1 Data cache misses	33.645M/sec	65 misses
LD & ST per D1 miss		20.51 ops/miss
D1 cache hit ratio		95.1%
LD & ST per D2 miss		1333.00 ops/miss
D2 cache hit ratio		100.0%
L2 cache hit ratio		100.0%
Memory to D1 refill		0 lines
Memory to D1 bandwidth		0 bytes
L2 to Dcache bandwidth	2053.542MB/sec	4160 bytes

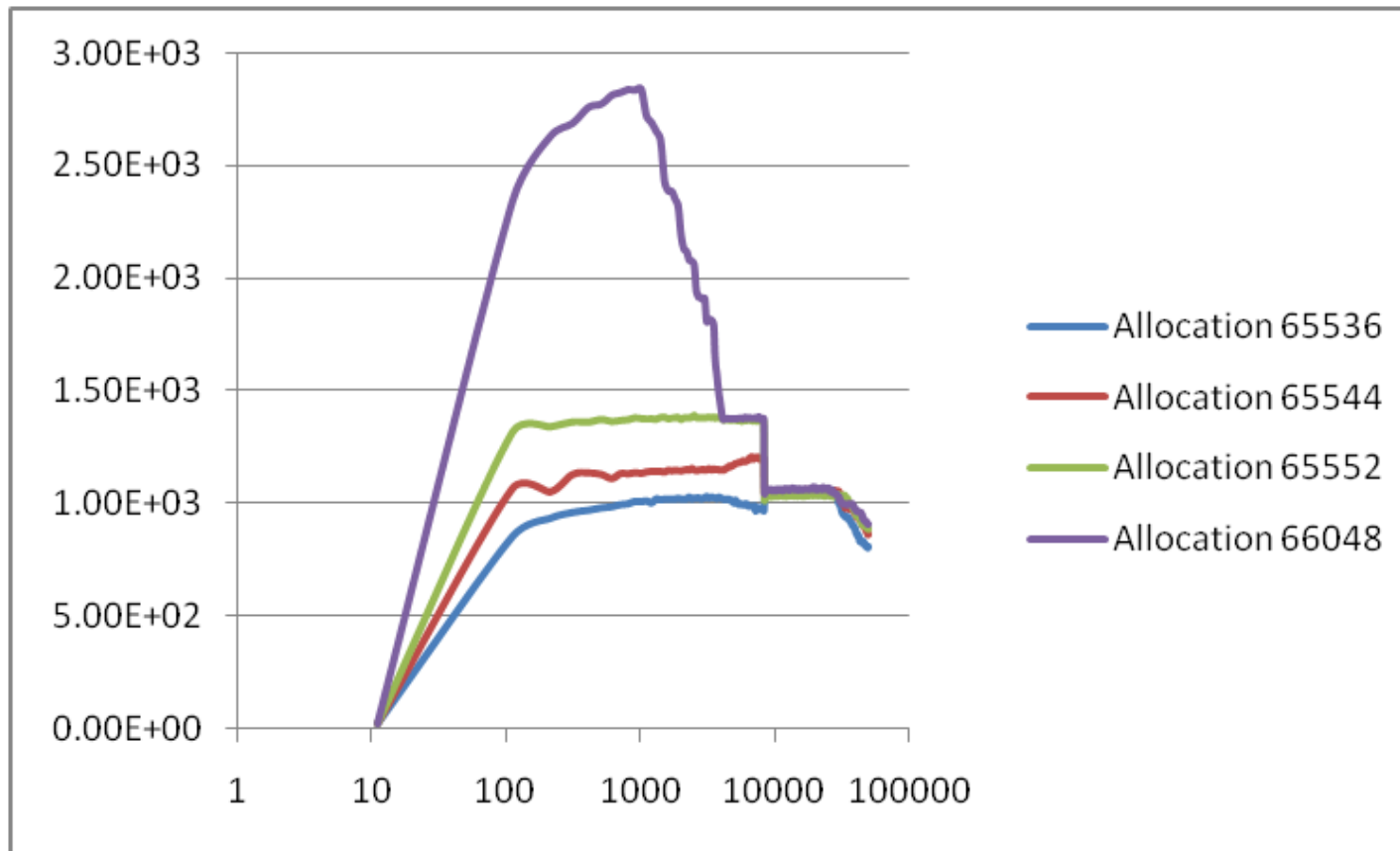
# Performance = F( Cache Utilization )

Stream Triad (MFLOPS)



# Performance = F( Cache Utilization )

Stream Triad (MFLOPS)



# NPB MG routine RESID

```
do i3=2,n3-1
  do i2=2,n2-1
    do i1=1,n1
      u1(i1) = u(i1,i2-1,i3) + u(i1,i2+1,i3)
>           + u(i1,i2,i3-1) + u(i1,i2,i3+1)
      u2(i1) = u(i1,i2-1,i3-1) + u(i1,i2+1,i3-1)
>           + u(i1,i2-1,i3+1) + u(i1,i2+1,i3+1)
    enddo
  do i1=2,n1-1
    r(i1,i2,i3) = v(i1,i2,i3)
>               - a(0) * u(i1,i2,i3)
>               - a(2) * ( u2(i1) + u1(i1-1) + u1(i1+1) )
>               - a(3) * ( u2(i1-1) + u2(i1+1) )
  enddo
enddo
enddo
7/17/09 enddo
```

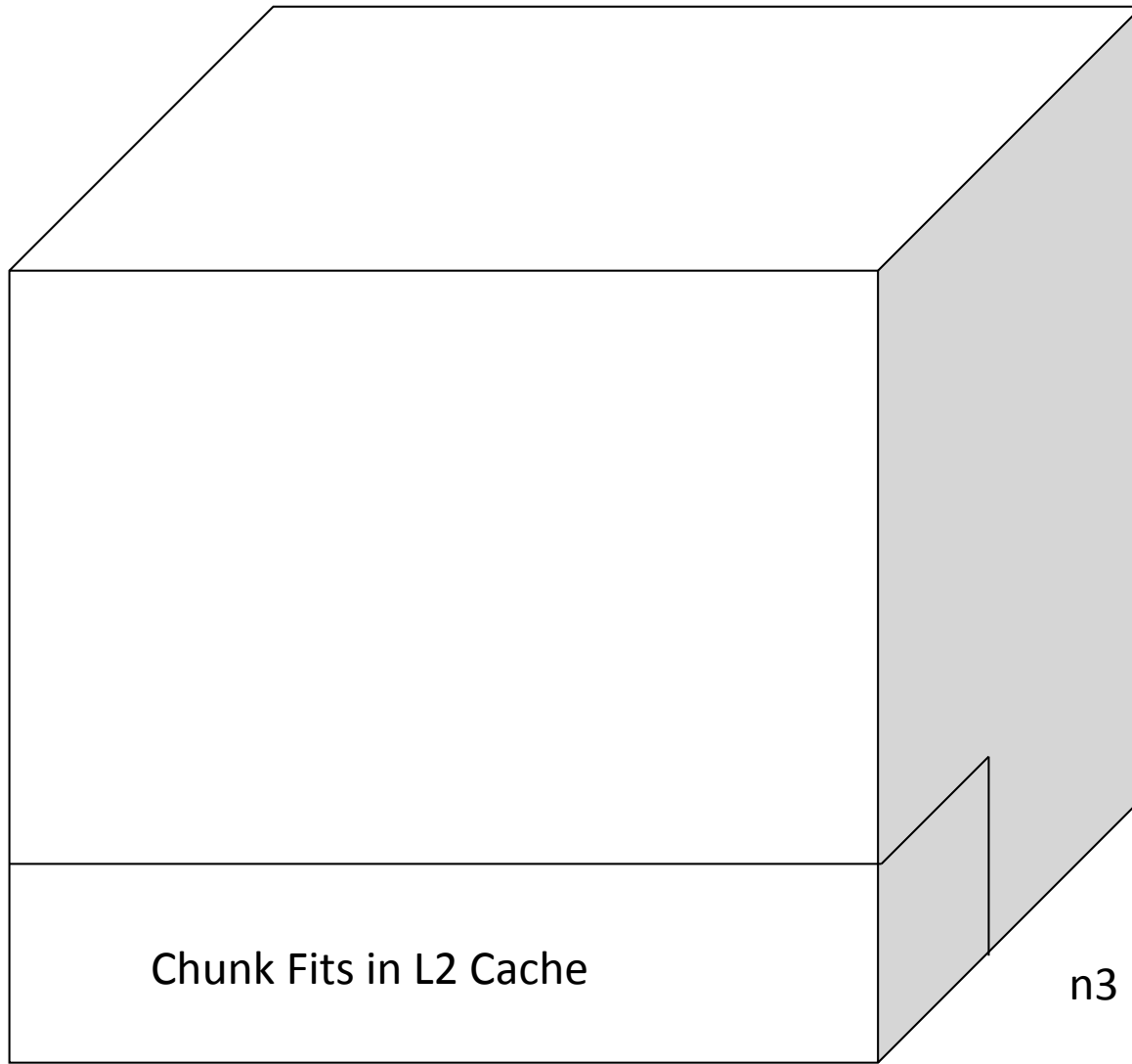
=====

USER / resid\_

-----

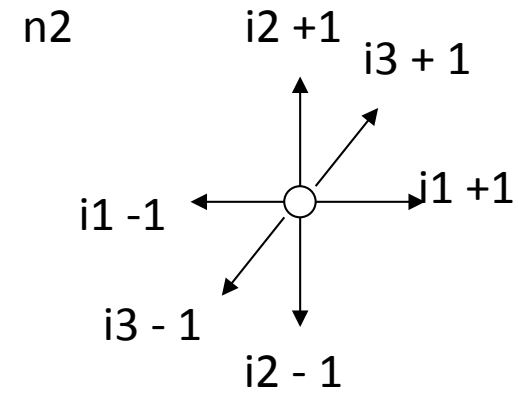
Time%		42.4%
Time		12.397761
Imb.Time		0.000370
Imb.Time%		0.0%
Calls		340
PAPI_L1_DCA	2719.188M/sec	33711498004 ops
DC_L2_REFILL_MOESI	79.644M/sec	987402929 ops
DC_SYS_REFILL_MOESI	4.059M/sec	50318116 ops
BU_L2_REQ_DC	129.172M/sec	1601429574 req
User time	12.398 secs	32233848320 cycles
Utilization rate		100.0%
L1 Data cache misses	83.703M/sec	1037721045 misses
LD & ST per D1 miss		32.49 ops/miss
D1 cache hit ratio		96.9%
LD & ST per D2 miss		669.97 ops/miss
D2 cache hit ratio		96.9%
L2 cache hit ratio		95.2%
Memory to D1 refill	4.059M/sec	50318116 lines
Memory to D1 bandwidth	247.723MB/sec	3220359424 bytes
L2 to Dcache bandwidth	4861.112MB/sec	63193787456 bytes

7/17/09



Entire Cube does not fit in L2 Cache

$256 * 256 * 256 * 3$  arrays  
 = 402 MBytes



Take data in chunks that Fit in L2 Cache

$256 * 16 * 32 * 3$  arrays  
 = 1 MBytes



# Tiling for better Cache utilization

```
do i3block=2,n3-1,BLOCK3
  do i2block=2,n2-1,BLOCK2
    do i3=i3block,min(n3-1,i3block+BLOCK3-1)
      do i2=i2block,min(n2-1,i2block+BLOCK2-1)
        do i1=1, n1
          u1(i1) = u(i1,i2-1,i3) + u(i1,i2+1,i3)
          >          + u(i1,i2,i3-1) + u(i1,i2,i3+1)
          u2(i1) = u(i1,i2-1,i3-1) + u(i1,i2+1,i3-1)
          >          + u(i1,i2-1,i3+1) + u(i1,i2+1,i3+1)
        enddo
        do i1=1, n1
          r(i1,i2,i3) = v(i1,i2,i3)
          >          - a(0) * u(i1,i2,i3)
          >          - a(2) * ( u2(i1) + u1(i1-1) + u1(i1+1) )
          >          - a(3) * ( u2(i1-1) + u2(i1+1) )
        enddo
      enddo
    enddo
  enddo
enddo
```

=====

USER / resid\_

-----

Time%		36.3%
Time		8.753226
Imb.Time		0.000596
Imb.Time%		0.0%
Calls		340
PAPI_L1_DCA	3861.533M/sec	33800955933 ops
DC_L2_REFILL_MOESI	116.399M/sec	1018867620 ops
DC_SYS_REFILL_MOESI	2.755M/sec	24114222 ops
BU_L2_REQ_DC	161.490M/sec	1413560527 req
User time	8.753 secs	22758444048 cycles
Utilization rate		100.0%
L1 Data cache misses	119.154M/sec	1042981842 misses
LD & ST per D1 miss		32.41 ops/miss
D1 cache hit ratio		96.9%
LD & ST per D2 miss		1401.70 ops/miss
D2 cache hit ratio		98.3%
L2 cache hit ratio		97.7%
Memory to D1 refill	2.755M/sec	24114222 lines
Memory to D1 bandwidth	168.145MB/sec	1543310208 bytes
L2 to Dcache bandwidth	7104.420MB/sec	65207527680 bytes

7/17/09

```

do i3block=2,n3-1,BLOCK3
do i2block=2,n2-1,BLOCK2
do i3=i3block,min(n3-1,i3block+BLOCK3-1)
do i2=i2block,min(n2-1,i2block+BLOCK2-1)
do i1=1,n1
u1(i1) = u(i1,i2-1,i3) + u(i1,i2+1,i3)
>          + u(i1,i2,i3-1) + u(i1,i2,i3+1)
u2(i1) = u(i1,i2-1,i3-1) + u(i1,i2+1,i3-1)
>          + u(i1,i2-1,i3+1) + u(i1,i2+1,i3+1)
enddo
do i1=2,n1-1
r(i1,i2,i3) = v(i1,i2,i3)
>          - a(0) * u(i1,i2,i3)
>          - a(2) * ( u2(i1) + u1(i1-1) + u1(i1+1) )
>          - a(3) * ( u2(i1-1) + u2(i1+1) )
enddo
enddo
enddo
enddo
enddo

```

```

do i3block=2,n3-1,BLOCK3
  do i2block=2,n2-1,BLOCK2
    do i3=i3block,min(n3-1,i3block+BLOCK3-1)
      do i2=i2block,min(n2-1,i2block+BLOCK2-1)
        do i1=2,n1-1
          u21 = u(i1,i2-1,i3-1) + u(i1,i2+1,i3-1)
>          + u(i1,i2-1,i3+1) + u(i1,i2+1,i3+1)
          u21p1 = u(i1+1,i2-1,i3-1) + u(i1+1,i2+1,i3-1)
>          + u(i1+1,i2-1,i3+1) + u(i1+1,i2+1,i3+1)
          u21m1 = u(i1-1,i2-1,i3-1) + u(i1-1,i2+1,i3-1)
>          + u(i1-1,i2-1,i3+1) + u(i1-1,i2+1,i3+1)
          u11p1 = u(i1+1,i2-1,i3) + u(i1+1,i2+1,i3)
>          + u(i1+1,i2,i3-1) + u(i1+1,i2,i3+1)
          u11m1 = u(i1-1,i2-1,i3) + u(i1-1,i2+1,i3)
>          + u(i1-1,i2,i3-1) + u(i1-1,i2,i3+1)
          r(i1,i2,i3) = v(i1,i2,i3)
>          - a(0) * u(i1,i2,i3)
>          - a(2) * ( u21 + u11m1 + u11p1 )
>          - a(3) * ( u21m1 + u21p1 )
        enddo
      enddo
    enddo
  enddo
enddo

```

USER / resid\_  
-----  
-----

Time%			37.7%
Time			9.132935
Imb.Time			0.003440
Imb.Time%			0.1%
Calls			340
PAPI_TLB_DM	0.139M/sec	1270096	misses
PAPI_L1_DCA	3694.219M/sec	33739238309	ops
PAPI_FP_OPS	2601.948M/sec	23763548027	ops
DC_MISS	111.833M/sec	1021371774	ops
User time	9.133 secs	23745753175	cycles
Utilization rate			100.0%
HW FP Ops / Cycles			1.00 ops/cycle
HW FP Ops / User time	2601.948M/sec	23763548027	ops
25.0%peak			
HW FP Ops / WCT	2601.948M/sec		
Computation intensity			0.70 ops/ref
LD & ST per TLB miss		26564.32	ops/miss
LD & ST per D1 miss		33.03	ops/miss
D1 cache hit ratio			97.0%
% TLB misses / cycle			0.0%

USER / resid\_  
-----  
---

Time%			39.6%	
Time			9.752716	
Imb.Time			0.002081	
Imb.Time%			0.0%	
Calls			340	
PAPI_TLB_DM	0.115M/sec		1119418	misses
PAPI_L1_DCA	2792.319M/sec		27232706384	ops
PAPI_FP_OPS	3488.881M/sec		34026076279	ops
DC_MISS	104.718M/sec		1021283533	ops
User time	9.753 secs		25357072370	cycles
Utilization rate			100.0%	
HW FP Ops / Cycles			1.34	ops/cycle
HW FP Ops / User time	3488.881M/sec		34026076279	ops 33.5%peak
HW FP Ops / WCT	3488.881M/sec			
Computation intensity			1.25	ops/ref
LD & ST per TLB miss			24327.56	ops/miss
LD & ST per D1 miss			26.67	ops/miss
D1 cache hit ratio			96.2%	
% TLB misses / cycle			0.0%	

USER / resid\_

-----  
 -----

Time%		38.3%	
Time		9.162149	
Imb.Time		0.006363	
Imb.Time%		0.1%	
Calls		340	
PAPI_L1_DCA	3682.405M/sec	33739250204	ops
DC_L2_REFILL_MOESI	111.475M/sec	1021369289	ops
DC_SYS_REFILL_MOESI	2.964M/sec	27157915	ops
BU_L2_REQ_DC	157.164M/sec	1439982850	req
User time	9.162 secs	23821945786	cycles
Utilization rate		100.0%	
L1 Data cache misses	114.439M/sec	1048527204	misses
LD & ST per D1 miss		32.18	ops/miss
D1 cache hit ratio		96.9%	
LD & ST per D2 miss		1242.34	ops/miss
D2 cache hit ratio		98.1%	
L2 cache hit ratio		97.4%	
Memory to D1 refill	2.964M/sec	27157915	lines
Memory to D1 bandwidth	180.914MB/sec	1738106560	bytes
L2 to Dcache bandwidth	6803.916MB/sec	65367634496	bytes

USER / resid\_

-----  
 -----

Time%			39.4%
Time			9.699533
Imb.Time			0.003564
Imb.Time%			0.1%
Calls			340
PAPI_L1_DCA	2807.643M/sec	27232738768	ops
DC_L2_REFILL_MOESI	105.292M/sec	1021281565	ops
DC_SYS_REFILL_MOESI	2.366M/sec	22945693	ops
BU_L2_REQ_DC	114.970M/sec	1115152062	req
User time	9.700 secs	25218702347	cycles
Utilization rate			100.0%
L1 Data cache misses	107.658M/sec	1044227258	misses
LD & ST per D1 miss		26.08	ops/miss
D1 cache hit ratio		96.2%	
LD & ST per D2 miss		1186.83	ops/miss
D2 cache hit ratio		97.9%	
L2 cache hit ratio		97.8%	
Memory to D1 refill	2.366M/sec	22945693	lines
Memory to D1 bandwidth	144.388MB/sec	1468524352	bytes
L2 to Dcache bandwidth	6426.524MB/sec	65362020160	bytes



# MHD3D Original

```
DO 200 K=0, KX
DO 200 J=0, JX
DO 200 I=0, IX
    F(I, J, K) = RVX(I, J, K)
    G(I, J, K) = RVY(I, J, K)
    H(I, J, K) = RVZ(I, J, K)
    S(I, J, K) = 0.
200 CONTINUE
CALL HALF(RO, ROH, DRO, F, G, H, S)
```

# Original HALF

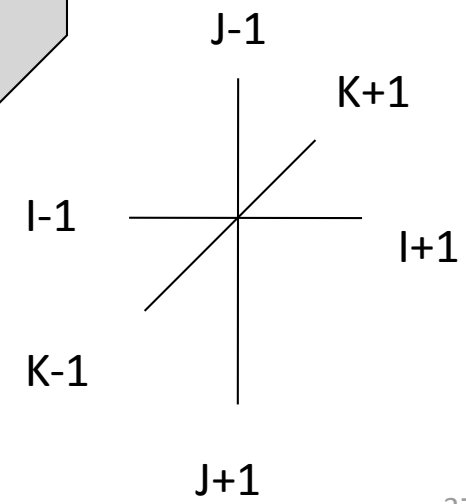
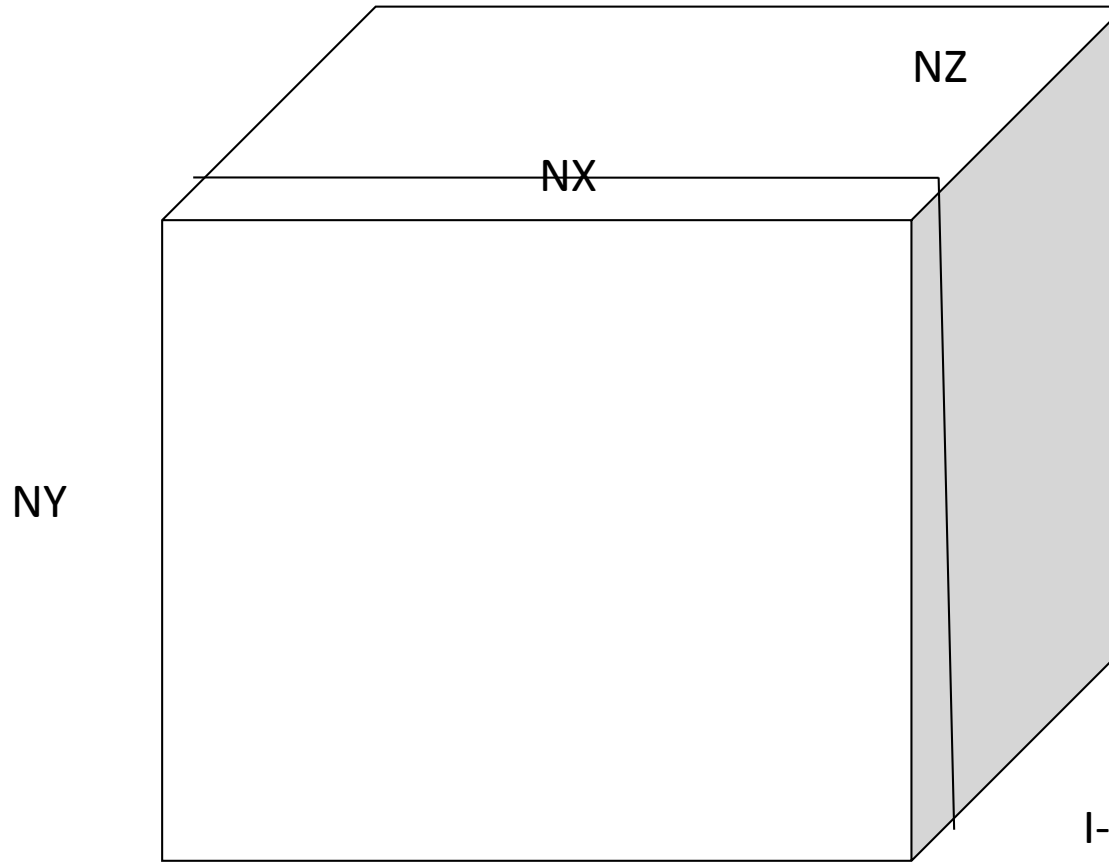
```
C=====
      DO 100 K=1,KXS1
      DO 100 J=1,JXS1
      DO 100 I=1,IXS1
          DU(I,J,K)=DU(I,J,K)-0.5*DT*
&          (0.5*RDXM(I)*(F(I+1,J,K)-F(I-1,J,K))
&          +0.5*RDYM(J)*(G(I,J+1,K)-G(I,J-1,K))
&          +0.5*RDZM(K)*(H(I,J,K+1)-H(I,J,K-1))
&          +S(I,J,K))
100    CONTINUE
C=====
C***  proceed half step using flux across cell boundary      ***
C=====
```

# Original HALF

```
DO 200 K=0,KXS1
DO 200 J=0,JXS1
DO 200 I=0,IXS1
C----- cell average -----
      UH =0.125*(U(I+1,J+1,K+1)+U(I,J+1,K+1)
&          +U(I+1,J+1,K)  +U(I,J+1,K)
&          +U(I+1,J,K+1)  +U(I,J,K+1)
&          +U(I+1,J,K)    +U(I,J,K))
      SH =0.125*(S(I+1,J+1,K+1)+S(I,J+1,K+1)
&          +S(I+1,J+1,K)  +S(I,J+1,K)
&          +S(I+1,J,K+1)  +S(I,J,K+1)
&          +S(I+1,J,K)    +S(I,J,K))
C----- flux across cell boundary -----
      DFDX = 0.25*RDY(J)*(F(I+1,J+1,K+1)-F(I, J+1,K+1)
&          +F(I+1,J+1,K)  -F(I, J+1,K)
&          +F(I+1,J, K+1)-F(I, J, K+1)
&          +F(I+1,J, K)  -F(I, J, K))
      DGDY = 0.25*RDZ(K)*(G(I+1,J+1,K+1)-G(I+1,J, K+1)
&          +G(I+1,J+1,K)  -G(I+1,J, K)
&          +G(I, J+1,K+1)-G(I, J, K+1)
&          +G(I, J+1,K)  -G(I, J, K))
      DHDZ = 0.25*RDZ(K)*(
&          H(I+1,J+1,K+1)-H(I+1,J+1,K)
&          +H(I+1,J, K+1)-H(I+1,J, K)
&          +H(I, J+1,K+1)-H(I, J+1,K)
&          +H(I, J, K+1)-H(I, J, K))
C----- summation of all terms -----
      UN(I,J,K) = UH-DT*(DFDX+DGDY+DHDZ+SH)
200 CONTINUE
7/17/09 RETURN
END
```

# Storage Analysis

Original								
	Variables	NX	NY	NZ	Mwords	MB	L2	TLBs
Loop 200	7	259	255	9	20.8	37	75	38
Half Do 100	5	259	255	9	2.972025	11.8881	23.7762	11
Half Do 200	6	259	255	9	3.56643	14.26572	28.53144	14



HALF needs  $I+1, J+1, K+1$ , So MLWXYZ must  
 Update a K plane ahead prior to call to HALF

# MHD3D Restructured

```
DO K = 0, KX
  KDOWN=K+1
  KUP=K+1
  IF (K.EQ.0) THEN
    KDOWN=k
    KUP=k+1
  ENDIF
  IF (K.EQ.KX) THEN
    KDOWN=K+1
    KUP=K
  ENDIF
DO JJ = 0, JX, JBLOCK
  JSTART = JJ
  JSTOP = MIN(JSTART+JBLOCK, JX)
  IF (JJ.NE.0) THEN
    JSTART=JSTART+1
  ENDIF
```

# MDH3D Restructured

```
DO KK=KDOWN, KUP
DO 200 J=JSTART, JSTOP
DO 200 I=0, IX
    F(I, J, KK) = RVX(I, J, KK)
    G(I, J, KK) = RVY(I, J, KK)
    H(I, J, KK) = RVZ(I, J, KK)
    S(I, J, KK) = 0.
200 CONTINUE
ENDDO
CALL HALF(JSTART, JSTOP, K, RO, ROH, DRO, F, G, H, S, 0)
```

# RESTRUCTURED HALF

```
IF (K.GT.0.AND.K.LE.KXS1) THEN
DO 100 J=MAX(1,JSTART),MIN(JXS1,JSTOP)
DO 100 I=1,IXS1
    DU(I,J,K)=DU(I,J,K)-0.5*DT*
&          (0.5*RDXM(I)*(F(I+1,J,K)-F(I-1,J,K))
&          +0.5*RDYM(J)*(G(I,J+1,K)-G(I,J-1,K))
&          +0.5*RDZM(K)*(H(I,J,K+1)-H(I,J,K-1))
&          +S(I,J,K))
100 CONTINUE
ENDIF
C=====
C*** proceed half step using flux across cell boundary ***
C=====
```



# RESTRUCTURED HALF

```

IF (K.LT.KX) THEN
DO 200 J=MAX(0,JSTART),MIN(JXS1,JSTOP)
DO 200 I=0,IXS1
C----- cell average -----
      UH =0.125*(U(I+1,J+1,K+1)+U(I,J+1,K+1)
&          +U(I+1,J+1,K)  +U(I,J+1,K)
&          +U(I+1,J,K+1)  +U(I,J,K+1)
&          +U(I+1,J,K)    +U(I,J,K))
      SH =0.125*(S(I+1,J+1,K+1)+S(I,J+1,K+1)
&          +S(I+1,J+1,K)  +S(I,J+1,K)
&          +S(I+1,J,K+1)  +S(I,J,K+1)
&          +S(I+1,J,K)    +S(I,J,K))
C----- flux across cell boundary -----
      DFDX = 0.25*RDY(I)*(F(I+1,J+1,K+1)-F(I,J+1,K+1)
&          +F(I+1,J+1,K)  -F(I,J+1,K)
&          +F(I+1,J,K+1)  -F(I,J,K+1)
&          +F(I+1,J,K)    -F(I,J,K))
      DGDY = 0.25*RDY(J)*(G(I+1,J+1,K+1)-G(I+1,J,K+1)
&          +G(I+1,J+1,K)  -G(I+1,J,K)
&          +G(I,J+1,K+1)  -G(I,J,K+1)
&          +G(I,J+1,K)    -G(I,J,K))
      DHDZ = 0.25*RDZ(K)*(
&          H(I+1,J+1,K+1)-H(I+1,J+1,K)
&          +H(I+1,J,K+1)-H(I+1,J,K)
&          +H(I,J+1,K+1)-H(I,J+1,K)
&          +H(I,J,K+1)-H(I,J,K))
C----- summation of all terms -----
      UN(I,J,K) = UH-DT*(DFDX+DGDY+DHDZ+SH)
200 CONTINUE

```

# Storage Analysis

Restructured								
	Variables	NX	NY	NZ	Mwords	MB	L2	TLB
Loop 200	7	259	32	2	.116	..935	2	..95
Half Do 100	5	259	32	2	0.08288	.66	1.32	.66
Half Do 200	6	259	32	2	0.099456	.79	1.6	.79