Experimental Measurements of a Real-Time Garbage Collection Architecture

William J. Schmidt  
Iowa State University

Kelvin Nilsen  
Iowa State University

Follow this and additional works at: http://lib.dr.iastate.edu/cs_techreports

Part of the Systems Architecture Commons, and the Theory and Algorithms Commons

Recommended Citation

Computer Science Technical Reports. Paper 175.  
http://lib.dr.iastate.edu/cs_techreports/175
Experimental Measurements of a Real-Time Garbage Collection Architecture

TR 92-26
William J. Schmidt and Kelvin Nilsen

August 1992

Iowa State University of Science and Technology
Department of Computer Science
226 Atanasoff
Ames, IA 50011
Experimental Measurements of a Real-Time Garbage Collection Architecture

William J. Schmidt
Kelvin Nilsen

Department of Computer Science
Iowa State University

Abstract

This report contains the raw empirical data on which the results of ISU DCS Technical Report 92-25 are based. Section 1 contains the input data for the programs that constituted the experimental workload, and section 2 lists the configuration data for the simulated architecture. Sections 3, 4, and 5 contain the results of individual experiments used in chapters 5, 6, and 7, respectively, of TR 92-25.
Contents

1 Input Data for Test Cases .......................................................................................................................... 1
  1.1 Input data for sfift test cases .................................................................................................................. 1
  1.2 Input data for lisp test cases .................................................................................................................. 3
     1.2.1 db.lisp .............................................................................................................................................. 3
     1.2.2 prune.lisp ...................................................................................................................................... 6
  1.3 Input data for troff test cases ............................................................................................................... 7

2 Simulator Configurations .......................................................................................................................... 8
  2.1 Simulator configuration using a standard CPU ....................................................................................... 8
  2.2 Simulator configuration using a faster CPU ........................................................................................... 10

3 Raw Test Results for Chapter 5 Experiments ......................................................................................... 11
  3.1 gc/sfift/small/slowcpu/lowmem ........................................................................................................... 11
  3.2 gc/sfift/small/slowcpu/highmem .......................................................................................................... 14
  3.3 gc/sfift/small/fastcpu/lowmem ............................................................................................................ 16
  3.4 gc/sfift/small/fastcpu/highmem ........................................................................................................... 18
  3.5 gc/sfift/medium/slowcpu/lowmem ....................................................................................................... 21
  3.6 gc/sfift/medium/slowcpu/highmem ..................................................................................................... 23
  3.7 gc/sfift/medium/fastcpu/lowmem ........................................................................................................ 26
  3.8 gc/sfift/medium/fastcpu/highmem ....................................................................................................... 28
  3.9 gc/lisp/db/slowcpu/lowmem ............................................................................................................... 30
  3.10 gc/lisp/db/slowcpu/highmem .............................................................................................................. 33
  3.11 gc/lisp/db/fastcpu/lowmem ............................................................................................................... 35
  3.12 gc/lisp/db/fastcpu/highmem ............................................................................................................... 38
  3.13 gc/lisp/prune/slowcpu/lowmem ......................................................................................................... 40
  3.14 gc/lisp/prune/slowcpu/highmem ....................................................................................................... 43
  3.15 gc/lisp/prune/fastcpu/lowmem .......................................................................................................... 45
  3.16 gc/lisp/prune/fastcpu/highmem ......................................................................................................... 47
  3.17 gc/troff/osmpaper/slowcpu/lowmem ................................................................................................. 50
  3.18 gc/troff/osmpaper/slowcpu/highmem ............................................................................................... 52
  3.19 gc/troff/osmpaper/fastcpu/lowmem .................................................................................................. 55
  3.20 gc/troff/osmpaper/fastcpu/highmem ................................................................................................. 57
  3.21 gc/troff/toplas/slowcpu/lowmem ...................................................................................................... 60
  3.22 gc/troff/toplas/slowcpu/highmem .................................................................................................... 62
  3.23 gc/troff/toplas/fastcpu/lowmem ....................................................................................................... 64
  3.24 gc/troff/toplas/fastcpu/highmem ...................................................................................................... 67
  3.25 nogc/sfift/small/slowcpu .................................................................................................................... 69
  3.26 nogc/sfift/small/fastcpu ...................................................................................................................... 72
  3.27 nogc/sfift/medium/slowcpu ............................................................................................................... 74
  3.28 nogc/sfift/medium/fastcpu ................................................................................................................. 77
  3.29 nogc/lisp/db/slowcpu ......................................................................................................................... 79
  3.30 nogc/lisp/db/fastcpu ......................................................................................................................... 82
  3.31 nogc/lisp/prune/slowcpu .................................................................................................................... 85
3.32 noge/lisp/prune/fastcpu ......................................................... 87
3.33 noge/troff/osmpaper/slowcpu ...................................................... 90
3.34 noge/troff/osmpaper/fastcpu ....................................................... 92
3.35 noge/troff/toplas/slowcpu ......................................................... 95
3.36 noge/troff/toplas/fastcpu ......................................................... 97
3.37 Partial invalidation: sfft .......................................................... 100
3.38 Partial invalidation: lisp .......................................................... 103
3.39 Partial invalidation: troff .......................................................... 105
3.40 Scaling experiment: 0x40000 ....................................................... 108
3.41 Scaling experiment: 0x80000 ....................................................... 110
3.42 Scaling experiment: 0x100000 ..................................................... 113
3.43 Scaling experiment: 0x200000 ..................................................... 115
3.44 Scaling experiment: 0x40000 ....................................................... 118

4 Raw Test Results for Chapter 6 Experiments
4.1 Experimental results for SS compiler .............................................. 120
  4.1.1 sfft/small ................................................................. 120
  4.1.2 sfft/medium ............................................................ 123
  4.1.3 lisp/db ................................................................. 125
  4.1.4 lisp/prune ............................................................. 128
4.2 Experimental results for SU compiler .............................................. 130
  4.2.1 sfft/small ................................................................. 130
  4.2.2 sfft/medium ............................................................ 133
  4.2.3 lisp/db ................................................................. 135
  4.2.4 lisp/prune ............................................................. 137
4.3 Experimental results for HS compiler ............................................ 140
  4.3.1 sfft/small ................................................................. 140
  4.3.2 sfft/medium ............................................................ 142
  4.3.3 lisp/db ................................................................. 145
  4.3.4 lisp/prune ............................................................. 147
4.4 Experimental results for HU compiler ............................................ 150
  4.4.1 sfft/small ................................................................. 150
  4.4.2 sfft/medium ............................................................ 152
  4.4.3 lisp/db ................................................................. 155
  4.4.4 lisp/prune ............................................................. 157
4.5 Experimental results for HC compiler ............................................ 160
  4.5.1 sfft/small ................................................................. 160
  4.5.2 lisp/prune ............................................................. 162

5 Raw Test Results for Chapter 7 Experiments ..................................... 165
5.1 The RJS library ................................................................. 165
5.2 libg++ ................................................................. 167
5.3 The slice implementation ....................................................... 170
5.3.1 0x100000 ................................................................. 170
5.3.2 0x200000 ................................................................. 172
5.3.3 0x400000 ................................................................. 175
1.1 Input Data for Sft Test Cases

The following hexadecimal dump of medium was generated using "cd -H."

The small and medium input files used for the sft test cases were generated by a program to store random eight patterns into a file. The small file contains 512 bytes, while the medium contains 2048 bytes. The following hexadecimal dump of small was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The following hexadecimal dump of medium was generated using "cd -H."

The followi
1.2 Input data for lisp test cases

1.2.1 db.lsp

The two short Lisp programs used as input to the lisp interpreter were contributed by Craig VanZante, who developed them as part of his duties as an NSF-sponsored undergraduate research assistant. The first of these, called db.lsp, is an extension of a relational database example taken from reference [1]. The database itself and three sample queries were added by Craig.

```lisp
(define union (s1 s2)
  (if (null? s1) s2)
```

3
(if (member? (car s1) s2)
  (union (cdr s1) s2)
  (cons (car s1) (union (cdr s1) s2))))

(define not (x) (if x '() 'T))
(set nullset '())
(define *+ (x) (+ x 1))
(define or (x y) (if x y))
(define atom? (x) (or (null? x) (or (number? x) (symbol? x))))
(define equal (l1 l2)
  (if (atom? l1) (= l1 l2)
    (if (atom? l2) '()
      (if (equal (car l1) (car l2))
        (equal (cdr l1) (cdr l2))
        '())))))

(define member? (x s)
  (if (null? s) '()
    (if (equal x (car s)) 'T (member? x (cdr s))))))

(define inter (s1 s2)
  (if (null? s1) s1
    (if (member? (car s1) s2)
      (cons (car s1) (inter (cdr s1) s2))
      (inter (cdr s1) s2))))

(define diff (s1 s2)
  (if (null? s1) s1
    (if (null? s2) s1
      (if (member? (car s1) s2)
        (diff (cdr s1) s2)
        (cons (car s1) (diff (cdr s1) s2))))))

(define UNION (r s)
  (if (not (equal (car r) (car s)))
    (print 'error)
    (cons (car r) (union (cdr r) (cdr s)))))

(define INTER (r s)
  (if (not (equal (car r) (car s)))
    (print 'error)
    (cons (car r) (inter (cdr r) (cdr s)))))

(define DIFF (r s)
  (if (not (equal (car r) (car s)))
    (print 'error)
    (cons (car r) (diff (cdr r) (cdr s)))))

(define SELECT (A v r)
  (cons (car r) (include-rows v (col-num A (car r)) (cdr r))))

(define col-num (A A-list)
  (if (= A (car A-list)) 0
    (+1 (col-num A (cdr A-list))))))

(define include-rows (v n rows)
  (if (null? rows) '()
    (if (= v (nth n (car rows)))
      (cons (car rows) (include-rows v n (cdr rows)))
      (include-rows v n (cdr rows))))))

(define nth (n l)
  (if (= n 0) (car l) (nth (- n 1) (cdr l))))

(define PROJECT (X r)
  (cons X (include-cols* (col-num* X (car r)) (cdr r))))

(define col-num* (X A-list)
  (if (null? X) '()
    (cons (col-num (car X) A-list) (col-num* (cdr X) A-list))))

(define include-cols* (col-nums rows)
  (if (null? rows) nullset
    (add1+ (include-cols col-nums (car rows))
      (include-cols* col-nums (cdr rows))))))

(define include-cols (col-nums row)
  (if (null? col-nums) '()
    (cons (nth (car col-nums) row)
      (include-cols (cdr col-nums) row))))
(define append (x y)
  (if (null? x) y (cons (car x) (append (cdr x) y))))
(define JOIN (x s)
  (begin
    (set intersection (inter (car r) (car s)))
    (set r-intersection (col-num* intersection (car r)))
    (set s-intersection (col-num* intersection (car s)))
    (set r-diff-s (diff (car r) intersection))
    (set r-diff-s-cols (col-num* r-diff-s (car r)))
    (set s-diff-r (diff (car s) intersection))
    (set s-diff-r-cols (col-num* s-diff-r (car s)))
    (cons (append intersection (append r-diff-s s-diff-r)
           (join-cols* r-intersection r-diff-s-cols s-intersection
                     s-diff-r-cols (cdr r) (cdr s))))))
(define join-cols* (X-r r-cols X-s s-cols r-rows s-rows)
  (begin
    (set new-rows '())
    (while (not (null? r-rows))
      (begin
        (set s-tmp s-rows)
        (while (not (null? s-tmp))
          (begin
            (if (equal (include-cols X-r (car r-rows))
                   (include-cols X-s (car s-tmp)))
                (set new-rows (cons (join-cols X-r r-cols s-cols
                                       (car r-rows) (car s-tmp))
                             (car s-tmp))))
            (set s-tmp (cdr s-tmp)))
          (set r-rows (cdr r-rows))))
    (set s-tmp s-rows)
    (while (not (null? s-tmp))
      (begin
        (if (equal (include-cols X-r (car r-rows))
                   (include-cols X-s (car s-tmp)))
                (set new-rows (cons (join-cols X-r r-cols s-cols
                                       (car r-rows) (car s-tmp))
                             (car s-tmp))))
            (set s-tmp (cdr s-tmp)))
    (set r-rows (cdr r-rows)))
    (append (include-cols X-r r-rows)
            (include-cols X-s s-rows)
            (include-cols r-cols r-row)
            (include-cols s-cols s-row)))
(define addelt (x s)
  (if (member? x x) s (cons x s)))
(define union (s1 s2)
  (if (null? s1) s2
      (if (member? (car s1) s2)
        (union (cdr s1) s2)
        (cons (car s1) (union (cdr s1) s2)))))
; define the tables in the database
(set LABS '(
(Dept Building # Brand SW HW Staff)
(ComS Atan 30 IBM s1 h1 Joe)
(Bio Bessey 15 Mac s2 h2 Mike)
(Psych Lago 20 IBM s2 h2 Jody)
(Math Carver 40 Mac s3 h3 Tamara)
(Eco Lago 12 IBM s2 h1 Shawn)
(Engl Ross 60 Mac s2 h2 Kelly)
(Eng Cooer 45 Mac s4 h4 Glenn)
(Eng Sweeney 15 DEC s5 h5 Liz)
(CC Durham 30 DEC s5 h5 Mike)
(CC Durham 50 Mac s3 h2 Liz))
(set FACILITIES '(
(Building Hours Location)
(Atan 24 NW)
(Bessey 8 NE)
(Lago 24 NE)
(Carver 10 SW)
(Ross 18 SE)
(Cooer 12 NW)
(Sweeney 8 NW)
(Durham 24 NW)))
(set SOFTWARE '(
(SW Wrdproc SS DB Special)
(s1 WordPerfect Lotus None None)
(s2 WorkPerfect Excel dBase None)
(s3 MSword Excel None Mathematica)
(s4 MSwork Excel None Canvas)
(s5 Emacs None None PAW)))
(set HARDWARE '(
(HW RAM HD Monitor)
(h1 4 80 m1)
(h2 3 40 m2)
(h3 8 30 m3)
(h4 5 80 m4)
(h5 16 300 m5)))
(set DISPLAYS '(
(Monitor Size Bits)
(m1 13 4)
(m2 16 8)
(m3 9 1)
(m4 13 8)
(m5 19 24)))
(set PERSONNEL '(
(Staff U/G Major Wage)
(Joe Under ComS 6.00)
(Mike Grad Bio 9.00)
(Jody Grad Psych 8.00)
(Tamara Under Math 7.00)
(Shawn Grad ComS 8.50)
(Kelly Under Accnt 7.50)
(Glenn Under CivE 8.00)
(Liz Under ChemE 7.50)))

; find the Building, #, Brand, and Location of all 24-hour labs
(PROJECT '(Building # Brand Location) (JOIN (SELECT 'Hours '24 FACILITIES) LABS))

; find all labs not staffed by graduates
(PROJECT '(Staff Building Brand)
 JOIN LABS (DIFF PERSONNEL (SELECT 'U/G 'Grad PERSONNEL)))

; find labs staffed by Kelly or Liz
(UNION (SELECT 'Staff 'Kelly LABS) (SELECT 'Staff 'Liz LABS))

1.2.2 prune.lsp

The second input program, prune.lsp, performs alpha-beta pruning on a game tree, returning the value of the best path through the tree. Two relatively large ternary game trees are used for the test.

; pruner
; implementation of alpha-beta pruning routine, returns the value for the
; "best" path
; Craig A VanZante
;
; l : list containing the tree with integer values as leaves
; a : lower bound alpha, initially set at -1000
; b : upper bound beta, b > a, initially set at 1000
; m : array of integers indicating whether parent of current child is MAX (1) or MIN (-1)
; num : temp variable used to store number of children in current node
; tmp : number of child being processed
; previous definitions needed: length, atom?, or, +1
(define or (x y) (if x y))
(define atom? (x) (or (null? x) (or (number? x) (symbol? x))))
(define +1 (x) (+ x 1))
(define length (l) (if (null? l) 0 (+1 (length (cdr l)))))
(define max (x y)
  (if (> x y) x y))
(define min (x y)
  (if (< x y) x y))
(define pruner* (l a b m tmp num)
  (if (atom? l) 1
    (if (= 1 (length l)) (car l)
     (begin
       (set num (length (cdr l)))
       (set temp 0)
       (if (= m 1)
        (begin
          (while (< tmp num); loop through children
           (begin
             (set a (max a (pruner* (if (= 1 (length l)) (car l) (car(cdr l))))
              a b (* -1 m) 0))
             (if (or (> a b) (= a b)); decide if rest of children need to be checked
              (set tmp (+ 1 num)) ; break out of loop
              (begin
                (set a (min a (pruner* (if (= 1 (length l)) (car l) (car(cdr l))))
                   a b (* -1 m) 0))
                (if (or (< a b) (= b a))
                 (begin
                  (set temp (+ 1 tmp))
                  (set l (cdr l)))))))
     (begin
       (while (< tmp num)
        (begin
          (set a (max a (pruner* (if (= 1 (length l)) (car l) (car(cdr l))))
            a b (* -1 m) 0))
          (if (or (> a b) (= a b))
           (set temp (+ 1 num))
           (begin
             (set a (min a (pruner* (if (= 1 (length l)) (car l) (car(cdr l))))
                a b (* -1 m) 0))
             (if (or (< a b) (= b a))
              (begin
                (set temp (+ 1 tmp))
                (set l (cdr l))))))))))
  (define pruner (l)
    (if (null? l) '() (pruner* l -1000 1000 1 0)))
(pruner 'A (B (E (W N 53 14 27) (O 2 31 85) (P 30 11 67)) (F (Q 50 7 39) (R 29 16 79) (S 3 99 90)) (G (T 40 26 32) (U 29 82 64) (V 15 61 73)) (C (H W 75 38 23) (X 40 43 37) (Y 68 46 69)) (I (Z 5 28 80) (AA 24 78 39) (AB 63 59 7)) (J (AC 42 63 59) (AD 32 55 85) (AE 16 36 20)) (D (K (AF 32 51 81) (AG 6 79 19) (AH 81 19 63)) (L (AI 19 39 61) (AJ 40 44 41) (AK 72 69 72)) (M (AL 64 84 31) (AM 23 27 94) (AN 82 11 1))))
(pruner 'A (B (E (W N 35 41 72) (O 2 13 58) (P 03 11 76)) (F (Q 05 7 93) (R 92 61 97) (S 3 99 09)) (G (T 04 62 23) (U 92 28 46) (V 51 16 37)) (C (H W 57 83 32) (X 04 34 73) (Y 86 64 96)) (I (Z 5 82 08) (AA 42 87 93) (AB 36 95 7)) (J (AC 24 36 95) (AD 23 55 58) (AE 61 63 02)) (D (K (AF 23 15 18) (AG 6 97 91) (AH 18 91 36)) (L (AI 91 93 16) (AJ 04 44 44) (AK 27 96 27)) (M (AL 46 48 13) (AM 32 72 49) (AN 28 11 1))))
quit

1.3 Input data for troff test cases

The osmpaper input file for the troff test case was generated by taking a draft of an eight-page, two-column, ten-point paper and doubling its length by duplicating the body of the
text. The toplas input file is a draft of a 24-page, two-column, ten-point paper. Both are too lengthy to include here, but are available upon request.

2 Simulator Configurations

The dixsimgc simulator is highly tunable, containing over one hundred parameters that can be reconfigured to test different assumptions about the simulated hardware. This section describes the configurations used in the experiments described in chapters 5 through 7 of reference [2].

The output from each of the experiments includes the configuration data for the simulator used during that experiment. However, there are only two basic configurations that were used for these experiments, so duplicating this data for each test case would result in hundreds of pages of redundant information. Instead, these two configurations are described in this section. The first configures the system to assume all hardware is roughly equivalent to hardware that would be available at this writing. The second is identical to the first, with the exception that the CPU is now assumed to run twice as fast as the CPU in the first configuration. Since the CPU cycle is the basis of accounting, all parameters are doubled for the remaining hardware components. This means that the CPU runs twice as fast as the microprocessor in the garbage-collecting memory module.

The configurations listed here assume that the garbage collector contains 0x80000 bytes of garbage-collected memory. This is not the case for all of the experiments. The actual amount of garbage-collected memory used in a given experiment can be determined by examining the command line listed in the output from that experiment. Each such command line contains a string of the form dixsimgc.\text{n}, where \text{n} is the hexadecimal representation of the amount of garbage-collected memory used during the corresponding experiment. If the command line also includes the string fastcpu, this indicates that the second configuration was used. Otherwise the first configuration was used.

2.1 Simulator configuration using a standard CPU

DLX Floating Point configuration:
 Num\text{\&}ers: 1, Num\text{\&}iders: 1, Num\text{\&}ultipliers: 1
 AddLatency: 4, DivLatency: 20, Multi\text{\&}latency: 10

ICache:
 block size: 1, num blocks per line: 2, num lines: 4096

DCache:
 block size: 1, num blocks per line: 2, num lines: 4096

Traditional memory: 0x0 to 0x400000
 text segment starts at 0x100
 minimum run-time stack: 0x800 (in bytes)
 write buffer slots: 3
 DRAM row addressing bits: 0xffffff800
 Select DRAM row cost: 6
 Fetch DRAM column cost: 1
 Update DRAM column cost: 1
 Cost to buffer update: 1

GC Arbiter:
ReadWordOverhead: 0, WriteWordOverhead: 0
InitWordOverheadCost: 1, TendWordOverheadCost: 2
TendWordInc1Cost: 4, TendWordInc2Cost: 6
InvalidateCycles: 2

GC Arbiter – microprocessor interface:
CopyBlockOverhead: 3, CopyBlockIncrement: 4
CopyScanBlockOverhead: 3, CopyScanBlockIncrement: 4
ScanBlockOverhead: 3, ScanBlockInc1: 2, ScanBlockInc2: 2
ScanAnyBlockOverhead: 3, ScanAnyBlockInc1: 2, ScanAnyBlockInc2: 2
SkipScanBlockOverhead: 3, ReadWordOverhead: 4
WriteWordOverhead: 3, DoFlipOverhead: 5
ZapFromSpaceOverhead: 4, CreateObjectOverhead: 4
FindHeaderOverhead: 4, ClearTagOverhead: 4
IncScanBalanceOverhead: 2, StartGCOverhead: 2
StopGCOverhead: 2

GC Arbiter – application processor interface:
GCArbInitBlockOverhead: 3
GCArbInitBlockIncrement: 1
GCArbCopyBlockOverhead: 4
GCArbCopyBlockIncrement: 2
GCArbCopyPushOverhead: 6
GCArbCopyPushIncrement: 2
GCArbTendDescOverhead: 2
GCArbTendDoneOverhead: 2
GCArbAllocRecOverhead: 4
GCArbAllocInitRecOverhead: 4
GCArbAllocInitRecInc: 2
GCArbAllocSliceOverhead: 8
GCArbAllocSliceIncrement: 4
GCArbAllocSubSliceOverhead: 8
GCArbAllocStackOverhead: 6
GCArbStackPushOverhead: 6
GCArbStackPushIncrement: 2
GCArbStackPopOverhead: 6
GCArbWordReadOverhead: 1
GCArbWordWriteOverhead: 1
GCArbContentionPenalty: 5

Garbage Collected memory: 0x800000 to 0x880000
write buffers (per space): 3
DRAM row addressing bits: 0xffffffff
  Select DRAM row cost: 6
  Fetch DRAM column cost: 1
  Update DRAM column cost: 1
  Cost to buffer update: 1
    ZapCostOverhead: 5
    ZapCostPerWord: 1

OSM:
createObject buffers: 1
  LookupCost: 6, CreateCost: 8
  ClearCostOverhead: 5, ClearCostPerGranule: 1

The Garbage Collecting Microprocessor:
gcK: 16
StopGCCost: 10, FlipOverheadCost: 10, BeginScanCost: 5
CopyObjectOverheadCost: 10, CopySmallDataOverheadCost: 16
ScanObjectOverheadCost: 10, ScanDataSliceOverheadCost: 16
MakeControlBlockOverheadCost: 10, MakeControlBlockIncCost: 6
DoControlBlockOverheadCost: 10, DoControlBlockIncCost: 6
MakeSmallDataRegionOverhead: 10
2.2 Simulator configuration using a faster CPU

DLX Floating Point configuration:
- NumAdders: 1, NumDividers: 1, NumMultipliers: 1
- AddLatency: 4, DivLatency: 20, MultiLatency: 10

ICache:
- block size: 1, num blocks per line: 2, num lines: 4096

DCache:
- block size: 1, num blocks per line: 2, num lines: 4096

Traditional memory: 0x0 to 0x400000
- text segment starts at 0x100
- minimum run-time stack: 0x800 (in bytes)
- write buffer slots: 3

DRAM row addressing bits: 0xffffffff
- Select DRAM row cost: 12
- Fetch DRAM column cost: 2
- Update DRAM column cost: 2
- Cost to buffer update: 2

GC Arbiter:
- ReadWordOverhead: 0, WriteWordOverhead: 0
- InitWordOverheadCost: 2, TendWordOverheadCost: 4
- TendWordInc1Cost: 8, TendWordInc2Cost: 12
- InvalidateCycles: 4

GC Arbiter - microprocessor interface:
- CopyBlockOverhead: 6, CopyBlockIncrement: 8
- CopyScanBlockOverhead: 6, CopyScanBlockIncrement: 8
- ScanBlockOverhead: 6, ScanBlockInc1: 4, ScanBlockInc2: 4
- ScanAnyBlockOverhead: 6, ScanAnyBlockInc1: 4, ScanAnyBlockInc2: 4
- SkipScanBlockOverhead: 6, ReadWordOverhead: 8
- WriteWordOverhead: 6, DoFlipOverhead: 10
- ZapFromSpaceOverhead: 8, CreateObjectOverhead: 8
- FindHeaderOverhead: 8, ClearTagOverhead: 8
- IncScanBalanceOverhead: 4, StartGCOverhead: 4
- StopGCOverhead: 4

GC Arbiter - application processor interface:
- GCArbInitBlockOverhead: 6
- GCArbInitBlockIncrement: 2
- GCArbCopyBlockOverhead: 8
- GCArbCopyBlockIncrement: 4
- GCArbCopyPushOverhead: 12
- GCArbCopyPushIncrement: 4
- GCArbTendDescOverhead: 4
- GCArbTendDoneOverhead: 4
- GCArbAllocRecOverhead: 8
- GCArbAllocInitRecOverhead: 8
- GCArbAllocInitRecInc: 4
- GCArbAllocSliceOverhead: 16
- GCArbAllocSliceIncrement: 8
- GCArbAllocSubSliceOverhead: 16
- GCArbAllocStackOverhead: 12
- GCArbStackPushOverhead: 12
- GCArbStackPushIncrement: 4
- GCArbStackPopOverhead: 12
- GCArbWordReadOverhead: 2
- GCArbWordWriteOverhead: 2
- GCArbContentionPenalty: 10

Garbage Collected memory: 0x800000 to 0x880000
write buffers (per space): 3
DRAM row addressing bits: 0xffffffff
Select DRAM row cost: 12
Fetch DRAM column cost: 2
Update DRAM column cost: 2
Cost to buffer update: 2
ZapCostOverhead: 10
ZapCostPerWord: 2

OSM:
createObject buffers: 1
LookupCost: 12, CreateCost: 16
ClearCostOverhead: 10, ClearCostPerGramule: 2

The Garbage Collecting Microprocessor:
gcK: 16
StopGCCost: 20, FlipOverheadCost: 20, BeginScanCost: 10
CopyObjectOverheadCost: 20, CopySmallDataOverheadCost: 32
ScanObjectOverheadCost: 20, ScanDataSliceOverheadCost: 32
MakeControlBlockOverheadCost: 20, MakeControlBlockIncCost: 12
DoControlBlockOverheadCost: 20, DoControlBlockIncCost: 12
MakeSmallDataRegionOverhead: 20

3 Raw Test Results for Chapter 5 Experiments

This appendix contains the raw empirical data produced by the simulation experiments described in chapter 5 of reference [2]. Each test case is given a four- or five-part name. The first part is “gc” if the garbage-collected memory architecture was simulated, or “nogc” if the standard architecture was used. The second part is the name of the program simulated, and the third part is the name of the input data to the simulated test program. The fourth part is either “slowcpu” or “fastcpu”, depending on whether the simulated processor has roughly the speed of a current-generation processor or twice that power. The fifth part is either “lowmem” or “highmem” depending on the amount of garbage-collected memory used; this part is omitted for “nogc” test cases.

3.1 gc/sfft/small/slowcpu/lowmem

Command line: /usr/schmidt/gnu/gc/sim/dlxsimgc.80000 sfft __start ../data/small

System Id:
$Id: ident.hh,v 1.13 1992/06/15 16:30:22 kelvin Exp $
This simulation begun on Fri Jun 19 08:34:52 1992

Statistics:

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>total machine instructions executed:</td>
<td>72545317</td>
</tr>
<tr>
<td>cycles stalled for instruction fetch:</td>
<td>5455961</td>
</tr>
<tr>
<td>cycles stalled for memory operations:</td>
<td>34285564</td>
</tr>
<tr>
<td>cycles stalled following loads:</td>
<td>31667176</td>
</tr>
<tr>
<td>cycles stalled for floating point results:</td>
<td>1456967</td>
</tr>
<tr>
<td>cycles stalled for floating point processors:</td>
<td>16383</td>
</tr>
<tr>
<td>cycles stalled for branch-delay instruction fetches:</td>
<td>4060</td>
</tr>
<tr>
<td>cycles stalled for trap interfacing:</td>
<td>4647</td>
</tr>
<tr>
<td>total machine cycles executed:</td>
<td>145436075</td>
</tr>
<tr>
<td>total number of traps executed:</td>
<td>199</td>
</tr>
<tr>
<td>Arbiter</td>
<td>Operation Invocations</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td></td>
<td>Operations</td>
</tr>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendingDone</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendDesc</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDSSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td>------------</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>174746</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>174746</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>137972</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>137972</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x82dd38)

number of allocations unimpeded by GC:    3
total cycles required for GC:    0

bus utilization:    37.497%
utilization due to cache invalidation requests:    1.672%
icache hit rate ( 77985338 hits / 77991967 fetches):    99.992%
dcache hit rate ( 20368329 hits / 23695346 fetches):    85.959%

Simulation completed on Fri Jun 19 22:08:52 1992
CPU usage: 94.9 system, 11751.1 user
3.2 gc/sfft/small/slowcpu/highmem

Command line: dlxsimgc.100000 sfft --start small

System Id:
$Id: ident.hh,v 1.13 1992/06/15 16:30:22 kelvin Exp $
This simulation begun on Mon Jun 15 17:23:54 1992

Statistics:
  total machine instructions executed:  72545317
  cycles stalled for instruction fetch:  5455961
  cycles stalled for memory operations:  34285564
  cycles stalled following loads:  31667176
  cycles stalled for floating point results:  1456967
  cycles stalled for floating point processors:  16383
  cycles stalled for branch-delay instruction fetches:  4060
  cycles stalled for trap interfacing:  4647
  total machine cycles executed:  145436075
  total number of traps executed:  199

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>number</td>
</tr>
<tr>
<td>Operations</td>
<td></td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>345</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>345</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>71</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>71</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>allocRec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>allocDSlice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>allocTSlice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>allocDSSubSlice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>allocTSubSlice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>allocDStack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>allocTStack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>stackPush</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>stackPop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
</tbody>
</table>

*A value of -1 signifies that the mean is not available due to unclaimed results.*

warning: no garbage collection activity (New = 0x86ddd38)

| number of allocations unimpeded by GC | 3 |
| total cycles required for GC | 0 |
| bus utilization | 37.49% |
| utilization due to cache invalidation requests | 1.62% |
| icache hit rate (77985338 hits / 77991967 fetches) | 99.99% |
| dcache hit rate (20368329 hits / 23695346 fetches) | 85.95% |

Simulation completed on Tue Jun 16 03:47:51 1992
CPU usage: 20.2 system, 7409.5 user

### 3.3 gc/sft/small/fastcpu/lowmem

dlxsimgc release:
`$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $`

Command line: fastcpu/dlxsimgc.80000 sft --start small

This simulation begun on Sat Jun 27 09:40:19 1992

Statistics:
Program image ends at 0x35d30

total machine instructions executed: 76173705
cycles stalled for instruction fetch: 6382478
cycles stalled for memory operations: 5631408
cycles stalled following loads: 49025797
cycles stalled for floating point processors: 1456956
cycles stalled for branch-delay instruction fetches: 5588
cycles stalled for trap interfacing: 7553

--------------------------------------------------
total machine cycles executed: 189382668
total number of traps executed: 199

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
</tbody>
</table>

<p>| latencies: |  |
| copyBlock: |  |
| Latencies: |  |
| During GC: | 0 | 0 | 0 | 0 |
| While GC idle: | 345 | 127 | 54.2017 |  |
| Costs: |  |
| During GC: | 0 | 0 | 0 | 0 | 0 |
| While GC idle: | 345 | 123.029 | 51.4832 | 57 | 177 |
| tendingDone: |  |
| Latencies: |  |</p>
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
<th>Latencies:</th>
<th>Costs:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocInitRec</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>copyPush</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>initBlock</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>71</td>
<td>201.451</td>
<td>26.0663</td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>71</td>
<td>197.789</td>
<td>25.7138</td>
<td>27</td>
</tr>
<tr>
<td><strong>tendDesc</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocRec</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>2</td>
<td>49</td>
<td>19.799</td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>2</td>
<td>21</td>
<td>0</td>
<td>21</td>
</tr>
<tr>
<td><strong>allocDSSlice</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTSlice</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocDSSubSlice</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTSSubSlice</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Costs:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
<th>allocDStack</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Costs:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

stackPush
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Costs:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

stackPop
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Costs:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x82dd38)

number of allocations unimpeded by GC: 3
total cycles required for GC: 0

bus utilization: 47.97%
utilization due to cache invalidation requests: 3.853%
icache hit rate ( 82520819 hits / 82527451 fetches): 99.992%
dcache hit rate ( 20351854 hits / 24602443 fetches): 82.723%

Simulation completed on Sat Jun 27 20:20:45 1992
CPU usage: 17.8 system, 8466.2 user

3.4 gc/sfft/small/fastcpu/highmem
dlxsimgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: fastcpu/dlxsimgc.100000 sfft __start small

This simulation begun on Sat Jun 27 09:40:48 1992

Statistics:
  Program image ends at 0x35d30

18
total machine instructions executed: 76173705
cycles stalled for instruction fetch: 6382478
cycles stalled for memory operations: 56314208
cycles stalled following loads: 49025797
cycles stalled for floating point results: 1466956
cycles stalled for floating point processors: 16363
cycles stalled for branch-delay instruction fetches: 5588
cycles stalled for trap interfacing: 7553

-----------------
total machine cycles executed: 189382668

-----------------total number of traps executed: 199

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>345</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>345</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>71</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>71</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------</td>
</tr>
<tr>
<td>allocDSlice</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.
warning: no garbage collection activity (New = 0x86dd38)

number of allocations unimpeded by GC: 3
total cycles required for GC: 0

bus utilization: 47.97%
utilization due to cache invalidation requests: 3.853%
icache hit rate ( 82520819 hits / 82527451 fetches): 99.992%
dcache hit rate ( 20351854 hits / 24602443 fetches): 82.723%

Simulation completed on Sat Jun 27 19:53:19 1992
CPU usage: 16.2 system, 8414.4 user

3.5 gc/sfft/medium/slowcpu/lowmem

Command line: /usr/schmidt/gnugc/sim/dlxsimgc.80000 sfft __start ../data/medium

System Id:
$Id: ident.hh,v 1.13 1992/06/15 16:30:22 kelvin Exp $
This simulation begun on Fri Jun 19 22:54:01 1992

Statistics:

  total machine instructions executed: 285568656
  cycles stalled for instruction fetch: 20791483
  cycles stalled for memory operations: 136019950
  cycles stalled following loads: 123936467
  cycles stalled for floating point results: 5773023
  cycles stalled for floating point processors: 65535
  cycles stalled for branch-delay instruction fetches: 4515
  cycles stalled for trap interfacing: 10023

  total machine cycles executed: 572169652
  total number of traps executed: 423

--------------------------------------------------------------------------
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>number</td>
</tr>
</tbody>
</table>
--------------------------------------------------------------------------
| copyBlock |         |       |         |        |
| Latencies:|         |       |         |        |
| During GC | 0      | 0     | 0       | 0      |
| While GC idle | 724 | 67.3812| 23.4152| 89     |
| Costs:    |         |       |         |        |
| During GC | 0      | 0     | 0       | 0      |
| While GC idle | 724 | 63.2818| 25.7525| 29     |
| tendingDone|        |       |         |        |
| Latencies:|         |       |         |        |
| During GC | 0      | 0     | 0       | 0      |
| While GC idle | 0 | 0     | 0       | 0      |
| Costs:    |         |       |         |        |
| During GC | 0      | 0     | 0       | 0      |
| While GC idle | 0 | 0     | 0       | 0      |
| allocInitRec|       |       |         |        |
| Latencies:|         |       |         |        |
| During GC | 0      | 0     | 0       | 0      |
| While GC idle | 0 | 0     | 0       | 0      |
| Costs:    |         |       |         |        |
| During GC | 0      | 0     | 0       | 0      |

21
<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>copyPush Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>71</td>
<td>169.225</td>
<td>16.5281</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>71</td>
<td>124.225</td>
<td>16.5953</td>
<td>14</td>
<td>127</td>
</tr>
<tr>
<td>tendDesc Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>2</td>
<td>13</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2</td>
<td>11</td>
<td>0</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>allocDSlice Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>13</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>12</td>
<td>0</td>
<td>12</td>
<td>12</td>
</tr>
</tbody>
</table>
Latencies: | | | | |
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 |

Costs: | | | |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |

stackPush: | | | |
Latencies: | | | |
| During GC | 0 | 0 | 0 |
| While GC idle | 649996 | 68.1785 | 34.657 |

Costs: | | | |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 649996 | 63.056 | 34.4446 | 15 | 127 |

stackPop: | | | |
Latencies: | | | |
| During GC | 0 | 0 | 0 |
| While GC idle | 540610 | 16.0309 | 0.526574 |

Costs: | | | |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 540610 | 14.2572 | 2.07082 | 9 | 23 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x82dd38)

number of allocations unimpeded by GC: 3
total cycles required for GC: 0

bus utilization: 37.65%
utilization due to cache invalidation requests: 1.476%
icache hit rate ( 306399068 hits / 306406380 fetches): 99.998%
dcache hit rate ( 81276890 hits / 93777223 fetches): 86.670%

Simulation completed on Mon Jun 22 06:09:56 1992
CPU usage: 294.4 system, 46047.0 user

3.6 gc/sfft/medium/slowcpu/highmem

Command line: d1xsimgc.100000 sfft __start medium

System Id:
$Id: ident.hh,v 1.13 1992/06/15 16:30:22 kelvin Exp $
This simulation begun on Mon Jun 15 14:21:43 1992

Statistics:

- total machine instructions executed: 285568656
- cycles stalled for instruction fetch: 20791483
- cycles stalled for memory operations: 136019950
- cycles stalled following loads: 123956467
- cycles stalled for floating point results: 5773023
- cycles stalled for floating point processors: 65535
- cycles stalled for branch-delay instruction fetches: 4515
- cycles stalled for trap interfacing: 10023

- total machine cycles executed: 572169652
- total number of traps executed: 423

---

23
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>724</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>724</td>
</tr>
</tbody>
</table>
tendingDone  |        |      |         |       |
| Latencies:   |        |      |         |       |
| During GC    | 0      | 0    | 0       | 0     |
| While GC idle| 0   | 0     | 0       | 0     |
| Costs:       |        |      |         |       |
| During GC    | 0      | 0    | 0       | 0     |
| While GC idle| 0   | 0     | 0       | 0     |
| allocInitRec |        |      |         |       |
| Latencies:   |        |      |         |       |
| During GC    | 0      | 0    | 0       |       |
| While GC idle| 0   | 0     | 0       |       |
| Costs:       |        |      |         |       |
| During GC    | 0      | 0    | 0       | 0     |
| While GC idle| 0   | 0     | 0       | 0     |
copyPush     |        |      |         |       |
| Latencies:   |        |      |         |       |
| During GC    | 0      | 0    | 0       | 0     |
| While GC idle| 0   | 0     | 0       | 0     |
| Costs:       |        |      |         |       |
| During GC    | 0      | 0    | 0       | 0     |
| While GC idle| 0   | 0     | 0       | 0     |
initBlock     |        |      |         |       |
| Latencies:   |        |      |         |       |
| During GC    | 0      | 0    | 0       |       |
| While GC idle| 71 | 129.225 | 16.5281 |       |
| Costs:       |        |      |         |       |
| During GC    | 0      | 0    | 0       | 0     |
| While GC idle| 71 | 124.225 | 16.5953 | 14    | 127   |
tendDesc     |        |      |         |       |
| Latencies:   |        |      |         |       |
| During GC    | 0      | 0    | 0       | 0     |
| While GC idle| 0   | 0     | 0       | 0     |
| Costs:       |        |      |         |       |
| During GC    | 0      | 0    | 0       | 0     |
| While GC idle| 0   | 0     | 0       | 0     |
allocRec     |        |      |         |       |
| Latencies:   |        |      |         |       |
| During GC    | 0      | 0    | 0       |       |
| While GC idle| 2  | 13    | 0       |       |
| Costs:       |        |      |         |       |
| During GC    | 0      | 0    | 0       | 0     |
| While GC idle| 2  | 11    | 0       | 11    | 11    |
allocDSlice  |        |      |         |       |
| Latencies:   |        |      |         |       |
| During GC    | 0      | 0    | 0       |       |
| While GC idle| 0   | 0     | 0       |       |
| Costs:       |        |      |         |       |
| During GC    | 0      | 0    | 0       | 0     |
| While GC idle| 0   | 0     | 0       | 0     |
allocTSlice  |        |      |         |       |
<p>| Latencies:   |        |      |         |       |
| During GC    | 0      | 0    | 0       |       |
| While GC idle| 0   | 0     | 0       |       |</p>
<table>
<thead>
<tr>
<th>Costs:</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>649996</td>
<td>68.1785</td>
<td>34.657</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>649996</td>
<td>63.056</td>
<td>34.4446</td>
<td>15</td>
<td>127</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>540610</td>
<td>16.0309</td>
<td>0.526574</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>540610</td>
<td>14.2572</td>
<td>2.07082</td>
<td>9</td>
<td>23</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x86dd38)

number of allocations unimpeded by GC: 3

bus utilization: 37.65%
utilization due to cache invalidation requests: 1.476%
icache hit rate ( 306399068 hits / 306406380 fetches): 99.998%
dcache hit rate ( 81276890 hits / 93777223 fetches): 86.670%

Simulation completed on Wed Jun 17 04:17:59 1992
CPU usage: 615.7 system, 29230.6 user
3.7 gc/sfft/medium/fastcpu/lowmem

dlxsimgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp$

Command line: fastcpu/dlxsimgc.80000 sfft --start medium

This simulation begun on Sat Jun 27 20:29:21 1992

Statistics:
Program image ends at 0x35d30

total machine instructions executed: 299094340
cycles stalled for instruction fetch: 24201957
cycles stalled for memory operations: 223593504
cycles stalled following loads: 189647151
cycles stalled for floating point results: 5773012
cycles stalled for floating point processors: 65535
cycles stalled for branch-delay instruction fetches: 6229
cycles stalled for trap interfacing: 16289

total machine cycles executed: 742388017

total number of traps executed: 423

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>------------------</td>
<td>--------</td>
</tr>
</tbody>
</table>

<p>| copyBlock        |        |       |         |       |
| Latencies:       |        |       |         |       |
| During GC        | 0      | 0     | 0       |       |
| While GC idle    | 724    | 129.677 | 54.233 |       |
| Costs:           |        |       |         |       |
| During GC        | 0      | 0     | 0       | 0     |
| While GC idle    | 724    | 125.564 | 51.505 | 57    | 177  |
| tendingDone      |        |       |         |       |
| Latencies:       |        |       |         |       |
| During GC        | 0      | 0     | 0       |       |
| While GC idle    | 0      | 0     | 0       |       |
| Costs:           |        |       |         |       |
| During GC        | 0      | 0     | 0       | 0     |
| While GC idle    | 0      | 0     | 0       | 0     |
| allocInitRec     |        |       |         |       |
| Latencies:       |        |       |         |       |
| During GC        | 0      | 0     | 0       |       |
| While GC idle    | 0      | 0     | 0       |       |
| Costs:           |        |       |         |       |
| During GC        | 0      | 0     | 0       | 0     |
| While GC idle    | 0      | 0     | 0       | 0     |
| copyPush         |        |       |         |       |
| Latencies:       |        |       |         |       |
| During GC        | 0      | 0     | 0       |       |
| While GC idle    | 0      | 0     | 0       |       |
| Costs:           |        |       |         |       |
| During GC        | 0      | 0     | 0       | 0     |
| While GC idle    | 0      | 0     | 0       | 0     |
| initBlock        |        |       |         |       |
| Latencies:       |        |       |         |       |
| During GC        | 0      | 0     | 0       |       |
| While GC idle    | 71     | 201.451 | 26.0663 |       |
| Costs:           |        |       |         |       |
| During GC        | 0      | 0     | 0       | 0     |</p>
<table>
<thead>
<tr>
<th>Method</th>
<th>Latencies:</th>
<th>Costs:</th>
<th>Latencies:</th>
<th>Costs:</th>
</tr>
</thead>
<tbody>
<tr>
<td>tendiff</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>2</td>
<td>49</td>
<td>0</td>
<td>21</td>
</tr>
<tr>
<td>Analyzed</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>35</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>23</td>
<td>0</td>
<td>23</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>64996</td>
<td>126.876</td>
<td>63.6189</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>64996</td>
<td>121.788</td>
<td>62.8552</td>
<td>29</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>64996</td>
<td>121.788</td>
<td>62.8552</td>
<td>29</td>
</tr>
</tbody>
</table>
Latencies: | | | | |
| During GC | 0 | 0 | 0 |
| While GC idle | 540610 | 32.9063 | 3.21082 |

Costs: | | | |
| During GC | 0 | 0 | 0 |
| While GC idle | 540610 | 27.5144 | 4.14163 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x82dd38)

- number of allocations unimpeded by GC: 3
- total cycles required for GC: 0
- bus utilization: 48.132%
- utilization due to cache invalidation requests: 3.413%
- icache hit rate (323306169 hits / 323313484 fetches): 99.996%
- dcache hit rate (81211170 hits / 97158644 fetches): 83.586%

CPU usage: 90.7 system, 33236.3 user

3.8 gc/sfft/medium/fastcpu/highmem
dlxsimc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: fastcpu/dlxsimc.100000 sfft __start medium

This simulation begun on Mon Jun 29 14:26:05 1992

Statistics:
- Program image ends at 0x35d30
  - total machine instructions executed: 29894340
  - cycles stalled for instruction fetch: 24201957
  - cycles stalled for memory operations: 223593504
  - cycles stalled following loads: 189647151
  - cycles stalled for floating point results: 5773012
  - cycles stalled for floating point processors: 65535
  - cycles stalled for branch-delay instruction fetches: 6229
  - cycles stalled for trap interfacing: 16289

  total machine cycles executed: 742398017

  total number of traps executed: 423

Arbiter | Operation Invocations
---------|-----------------------------------------
| number | mean* | std dev | range |
Operations | | | |
copyBlock | | | |

Latencies: | | | |
| During GC | 0 | 0 | 0 |
| While GC idle | 724 | 129.677 | 54.233 |

Costs: | | | |
<p>| During GC | 0 | 0 | 0 | 0 | 0 | 0 |</p>
<table>
<thead>
<tr>
<th></th>
<th>724</th>
<th>125.564</th>
<th>51.505</th>
<th>57</th>
<th>177</th>
</tr>
</thead>
<tbody>
<tr>
<td>while GC idle</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>tendingDone</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>allocDSlice</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Latencies: | | | | | |
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | |

Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

allocDStack Latencies: | | | |
| During GC | 0 | 0 | 0 | |
| While GC idle | 1 | 35 | 0 | |

Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 1 | 23 | 0 | 23 | 23 |

allocTStack Latencies: | | | |
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

stackPush Latencies: | | | |
| During GC | 0 | 0 | 0 | |
| While GC idle | 649996 | 126.876 | 63.6189 | |

Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 649996 | 121.788 | 62.8552 | 29 | 245 |

stackPop Latencies: | | | |
| During GC | 0 | 0 | 0 | |
| While GC idle | 540610 | 32.9063 | 3.21082 | |

Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 540610 | 27.5144 | 4.14163 | 17 | 45 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x86dd38)

number of allocations unimpeded by GC: 3
total cycles required for GC: 0

bus utilization: 48.13%
utilization due to cache invalidation requests: 3.413%
icache hit rate ( 323306169 hits / 323513484 fetches): 99.998%
dcache hit rate ( 81211170 hits / 97158644 fetches): 83.586%

Simulation completed on Wed Jul 1 12:32:35 1992
CPU usage: 103.1 system, 32986.0 user

3.9 gc/lisp/db/slowcpu/lowmem
dlxsimgc release:
Command line: slow/dlxsimgc.80000 lisp __start
This simulation begun on Thu Jul 2 11:35:39 1992

30
Statistics:
Program image ends at 0x2f6f0

<table>
<thead>
<tr>
<th></th>
<th>total machine instructions executed: 768918957</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>cycles stalled for instruction fetch: 158650914</td>
</tr>
<tr>
<td></td>
<td>cycles stalled for memory operations: 529600291</td>
</tr>
<tr>
<td></td>
<td>cycles stalled following loads: 471596326</td>
</tr>
<tr>
<td></td>
<td>cycles stalled for floating point results: 0</td>
</tr>
<tr>
<td></td>
<td>cycles stalled for floating point processors: 0</td>
</tr>
<tr>
<td></td>
<td>cycles stalled for branch-delay instruction fetches: 126654</td>
</tr>
<tr>
<td></td>
<td>cycles stalled for trap interfacing: 25453</td>
</tr>
<tr>
<td></td>
<td>total machine cycles executed: 1929008595</td>
</tr>
<tr>
<td></td>
<td>total number of traps executed: 948</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendingDone</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendDesc</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
</tbody>
</table>

31
| Latencies: | | | | |
|-----------|-----------|-----------|-----------|
| During GC | 0          | 0          | 0          |
| While GC idle | 776  19.1469 | 2.42441 |
| Costs: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 776  11.1418 | 0.830396 | 11  16 |
| allocDSlice |
| Latencies: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 0          | 0          |
| Costs: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 0          | 0          |
| allocTSlice |
| Latencies: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 0          | 0          |
| Costs: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 0          | 0          |
| allocDSSubSlice |
| Latencies: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 0          | 0          |
| Costs: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 0          | 0          |
| allocTSSubSlice |
| Latencies: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 0          | 0          |
| Costs: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 0          | 0          |
| allocDSStack |
| Latencies: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 1          | 13         |
| Costs: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 1          | 12         |
| allocTStack |
| Latencies: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 0          | 0          |
| Costs: | | | |
| During GC | 0          | 0          | 0          |
| While GC idle | 0          | 0          |
| stackPush |
| Latencies: | | | |
| During GC | 18027  43.4965 | 11.1097 |
| While GC idle | 9358611  40.6377 | 12.1873 |
| Costs: | | | |
| During GC | 18027  38.2927 | 10.3883 | 15  88 |
| While GC idle | 9358611  35.3974 | 11.4811 | 15  128 |
| stackPop |
| Latencies: | | | |
| During GC | 18181  25.6134 | 4.57908 |
| While GC idle | 9342713  21.5283 | 4.66694 |
| Costs: | | | |
| During GC | 18181  20.7711 | 3.69362 | 9  28 |
| While GC idle | 9342713  17.6314 | 3.564 | 9  28 |

ScanBalance/Cycles | 32670  0.365369 | 0.438995 | 0.00179558 | 3.05094 |
Heap Utilization | 4 | 0.00138618 | 0.000348511 | 0.00102234 | 0.00177002 |
3.10 gc/lisp/db/slowcpu/highmem

dlxsimgc release:  

Command line: slow/dlxsimgc.100000 lisp __start  

This simulation begun on Thu Jul 2 11:07:19 1992  

Statistics:  
Program image ends at 0x2f6f0  

total machine instructions executed: 748991516  
cycles stalled for instruction fetch: 153376905  
cycles stalled for memory operations: 430719526  
cycles stalled following loads: 419360908  
cycles stalled for floating point results: 0  
cycles stalled for floating point processors: 0  
cycles stalled for branch-delay instruction fetches: 123866  
cycles stalled for trap interfacing: 24463  

------------------------  
total machine cycles executed: 175259184  

------------------------  
total number of traps executed: 948

Arbiter | Operation Invocations
------- | ------------------------

<table>
<thead>
<tr>
<th>Operations</th>
<th>number</th>
<th>mean*</th>
<th>std dev</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td>copyBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.
<table>
<thead>
<tr>
<th>Latencies:</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td></td>
<td>22</td>
<td>44.2273</td>
<td>7.76397</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>28137</td>
<td>40.2289</td>
<td>7.45498</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>22</td>
<td>34.6818</td>
<td>3.95675</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>28137</td>
<td>33.4865</td>
<td>3.69879</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>820</td>
<td>20.9427</td>
<td>27.1744</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>820</td>
<td>17.7524</td>
<td>26.7308</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>13</td>
<td>37.0769</td>
<td>13.1559</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>13</td>
<td>19.1538</td>
<td>10.8693</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>776</td>
<td>19.1237</td>
<td>2.23639</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>776</td>
<td>11.0644</td>
<td>0.56429</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

34
While GC idle| 1| 13| 0| |
Costs: |
| 0| 0| 0| 0| 0| 0
While GC idle| 1| 12| 0| 12| 12
allocTStack |
Latencies: |
| 0| 0| 0| |
While GC idle| 0| 0| 0| |
Costs: |
| 0| 0| 0| 0| 0| 0
While GC idle| 0| 0| 0| 0| 0
stackPush |
Latencies: |
| 0| 0| 0| |
While GC idle| 5695| 43.0378| 11.2128| |
Costs: |
| 5695| 37.8376| 10.3839| 20| 83
While GC idle| 33.1765| 11.5691| 15| 128
stackPop |
Latencies: |
| 5673| 25.0673| 4.62917| |
While GC idle| 18.4951| 4.16772| |
Costs: |
| 5673| 20.3476| 3.64776| 9| 28
While GC idle| 15.3953| 3.72154| 9| 28
ScanBalance/Cycles| 10761| 0.535862| 2.06476| 0.00205488| 209.24
Heap Utilization| 1.0| 0.00846863| 0.00846863

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 28936

total cycles required for GC: 1277646

bus utilization: 32.836%

utilization due to cache invalidation requests: 2.165%
icache hit rate ( 899002040 hits / 899145912 fetches): 99.984%
dcache hit rate ( 36007919 hits / 134999835 fetches): 26.673%

Simulation completed on Sat Jul 4 12:52:44 1992
CPU usage: 554.5 system, 87537.8 user

3.11 gc/lisp/db/fastcpu/lowmem
dlxsimc release:

Command line: fast/dlxsimc.80000 lisp __start

This simulation begun on Thu Jul 2 11:06:31 1992

Statistics:
Program image ends at 0x2f6f0

total machine instructions executed: 951479694
cycles stalled for instruction fetch: 205445990
cycles stalled for memory operations: 897889349
cycles stalled following loads: 863093932
cycles stalled for floating point results: 0
cycles stalled for floating point processors: 0

35
cycles stalled for branch-delay instruction fetches: 173673
cycles stalled for trap interfacing: 42446

total machine cycles executed: 2918125084

total number of traps executed: 948

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>4</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>4</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>117</td>
</tr>
<tr>
<td>While GC idle</td>
<td>28045</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>117</td>
</tr>
<tr>
<td>While GC idle</td>
<td>28045</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>820</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>820</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>49</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>49</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>776</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>776</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
While GC idle | 0 | 0 | 0 |
Costs:  
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |
allocTSlice |  |
Latencies:  
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 |
Costs:  
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |
allocDSubSlice |  |
Latencies:  
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 |
Costs:  
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |
allocDStack |  |
Latencies:  
| During GC | 0 | 0 | 0 |
| While GC idle | 1 | 35 | 0 |
Costs:  
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 1 | 23 | 0 | 23 | 23 |
allocTSStack |  |
Latencies:  
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 |
Costs:  
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |
stackPush |  |
Latencies:  
| During GC | 32742 | 80.1046 | 22.208 |
| While GC idle | 9343896 | 74.6976 | 24.0915 |
Costs:  
| During GC | 32742 | 74.9368 | 20.793 | 29 | 183 |
| While GC idle | 9343896 | 69.7567 | 22.729 | 29 | 255 |
stackPop |  |
Latencies:  
| During GC | 32835 | 44.8324 | 7.90534 |
| While GC idle | 9328059 | 39.2576 | 6.1381 |
Costs:  
| During GC | 32835 | 39.9827 | 7.73385 | 17 | 55 |
| While GC idle | 9328059 | 34.2591 | 7.12923 | 17 | 55 |
Scanned Cycles / Cycles: 43162 | 0.330177 | 12.7625 | 0.000504032 | 2594
Heap Utilization: 4.00.00228119 | 0.000303785 | 0.00186157 | 0.00263977

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 28939

total cycles required for GC: 11445442

bus utilization: 44.176%
utilization due to cache invalidation requests: 3.903%
icache hit rate (1152043199 hits / 1152187618 fetches): 99.987%
dcache hit rate ( 3601470 hits / 185621872 fetches): 19.400%

Simulation completed on Sun Jul 5 10:27:31 1992
CPU usage: 982.7 system, 163583.7 user

3.12 gc/lisp/db/fastcpu/highmem
dlxsimc release:
Command line: fast/dlxsimc.100000 lisp _start
This simulation begun on Thu Jul 2 11:31:14 1992

Statistics:
   Program image ends at 0x2f6f0
   total machine instructions executed: 917598832
   cycles stalled for instruction fetch: 196381098
   cycles stalled for memory operations: 695747978
   cycles stalled following loads: 764371474
   cycles stalled for floating point results: 0
   cycles stalled for floating point processors: 0
   cycles stalled for branch-delay instruction fetches: 168941
   cycles stalled for trap interfacing: 40466
   ---------
   total machine cycles executed: 2574308789
   total number of traps executed: 948

<table>
<thead>
<tr>
<th>Operations</th>
<th>number</th>
<th>mean*</th>
<th>std dev</th>
<th>range</th>
</tr>
</thead>
</table>
| Arbiter          | Operation Invocations
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>1</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>37</td>
<td>76.6486</td>
<td>9.1906</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>28122</td>
<td>70.284</td>
<td>10.0687</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>37</td>
<td>69.3243</td>
<td>8.96305</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>28122</td>
<td>65.9705</td>
<td>7.39637</td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------</td>
<td>---------------</td>
<td>--------</td>
<td>------------</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>820</td>
<td>34.8012</td>
<td>42.4037</td>
<td>820</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>13</td>
<td>61.4615</td>
<td>21.9265</td>
<td>13</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
stackPush
Latencies:
During GC | 9651 | 79.4616 | 22.3044 |
While GC idle | 9366987 | 70.658 | 24.2028 |
Costs:
During GC | 9651 | 74.3441 | 20.7858 | 29 | 171 |
While GC idle | 9366987 | 65.3171 | 22.8888 | 29 | 255 |
stackPop
Latencies:
During GC | 9612 | 44.2132 | 7.55848 |
While GC idle | 9351282 | 35.2007 | 6.40115 |
Costs:
During GC | 9612 | 39.3908 | 7.51177 | 17 | 55 |
While GC idle | 9351282 | 29.7877 | 7.44247 | 17 | 55 |
ScanBalance/Cycles | 16005 | 0.382797 | 6.86802 | 0.000496032 | 868.75 |
Heap Utilization | 1 | 0.00414144 | 0 | 0.00414144 | 0.00414144 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

---

number of allocations unimpeded by GC: 28936
total cycles required for GC: 3316340
bus utilization: 39.644%
utilization due to cache invalidation requests: 4.421%
icache hit rate (1109703982 hits / 1109847858 fetches): 99.987%
dcache hit rate (36015969 hits / 177151664 fetches): 20.334%

Simulation completed on Tue Jul 7 07:04:55 1992
CPU usage: 1616.1 system, 257362.7 user

3.13 gc/lisp/prune/slowcpu/lowmem

dlxsimgc release:

Command line: slow/dlxsimgc.40000 lisp --start

This simulation begun on Mon Jun 29 14:46:40 1992

Statistics:
Program image ends at 0x2f6f0

  total machine instructions executed: 422534475
  cycles stalled for instruction fetch: 87610786
  cycles stalled for memory operations: 309091929
  cycles stalled following loads: 264915683
  cycles stalled for floating point results: 0
  cycles stalled for floating point processors: 0
  cycles stalled for branch-delay instruction fetches: 101361
  cycles stalled for trap interfacing: 1968

  total machine cycles executed: 1084256202

  total number of traps executed: 84

------------------------------------------------------------------------
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendDesc</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDSSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
</tbody>
</table>
During GC| 0| 0| 0| 0| 0 | 0
While GC idle| 0| 0| 0| 0| 0 | 0
allocDSSubSlice
Latencies:
| During GC| 0| 0| 0 | 0| 0 | 0
| While GC idle| 0| 0| 0 | 0| 0 | 0
Costs:
| During GC| 0| 0| 0 | 0| 0 | 0
| While GC idle| 0| 0| 0 | 0| 0 | 0
allocTSSubSlice
Latencies:
| During GC| 0| 0| 0 | 0| 0 | 0
| While GC idle| 0| 0| 0 | 0| 0 | 0
Costs:
| During GC| 0| 0| 0 | 0| 0 | 0
| While GC idle| 0| 0| 0 | 0| 0 | 0
allocDStack
Latencies:
| During GC| 0| 0| 0 | 0| 0 | 0
| While GC idle| 1| 13| 1 | 0| 0 | 0
Costs:
| During GC| 0| 0| 0 | 0| 0 | 0
| While GC idle| 1| 12| 0 | 12| 12 | 12
allocTStack
Latencies:
| During GC| 0| 0| 0 | 0| 0 | 0
| While GC idle| 0| 0| 0 | 0| 0 | 0
Costs:
| During GC| 0| 0| 0 | 0| 0 | 0
| While GC idle| 0| 0| 0 | 0| 0 | 0
stackPush
Latencies:
| During GC| 27475| 43.2002| 11.1674 | 11.1674 | 11.1674 | 11.1674
| While GC idle| 5083385| 41.0943| 11.8708 | 11.8708 | 11.8708 | 11.8708
Costs:
| During GC| 27475| 38.0631| 10.4405 | 10.4405 | 10.4405 | 10.4405
| While GC idle| 5083385| 35.9682| 10.8667 | 10.8667 | 10.8667 | 10.8667
stackPop
Latencies:
| During GC| 27496| 25.365| 4.6653 | 4.6653 | 4.6653 | 4.6653
| While GC idle| 5082036| 22.4602| 4.46188 | 4.46188 | 4.46188 | 4.46188
Costs:
| During GC| 27496| 20.588| 3.72072 | 3.72072 | 3.72072 | 3.72072
| While GC idle| 5082036| 18.3255| 3.26424 | 3.26424 | 3.26424 | 3.26424
ScanBalance/Cycles| 44126| 0.444735| 6.3093| 0.000400802 | 0.000400802 | 0.000400802
Heap Utilization| 12| 0.0013662| 0.000629719| 0.000640869 | 0.00256348

* A value of -1 signifies that the mean is not available due to unclaimed results.

---

number of allocations unimpeded by GC: 18809

total cycles required for GC: 6279160

bus utilization: 38.006%

utilization due to cache invalidation requests: 1.903%

icache hit rate (507729156 hits / 507838740 fetches): 99.978%

dcache hit rate (19494145 hits / 77113395 fetches): 25.280%

Simulation completed on Wed Jul 1 19:12:13 1992

CPU usage: 1200.2 system, 88188.9 user

---

42
3.14 gc/lisp/prune/slowcpu/highmem

dlxsimgc release:

Command line: slow/dlxsimgc.80000 lisp __start

This simulation begun on Thu Jul 2 11:31:52 1992

Statistics:
Program image ends at 0x2f6f0

| total machine instructions executed: | 414665368 |
| cycles stalled for instruction fetch: | 85482096 |
| cycles stalled for memory operations: | 269671239 |
| cycles stalled following loads: | 244374731 |
| cycles stalled for floating point results: | 0 |
| cycles stalled for floating point processors: | 0 |
| cycles stalled for branch-delay instruction fetches: | 97517 |
| cycles stalled for trap interfacing: | 1948 |
| total machine cycles executed: | 10142922899 |
| total number of traps executed: | 82 |

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operations Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>---------------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>28</td>
</tr>
<tr>
<td>While GC idle</td>
<td>18469</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>28</td>
</tr>
<tr>
<td>While GC idle</td>
<td>18469</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>345</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
</tbody>
</table>

43
<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td>26</td>
<td>13.389</td>
<td>3</td>
<td>8</td>
<td>127</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>26</td>
<td>10.8395</td>
<td>3</td>
<td>3</td>
<td>32</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>301</td>
<td>3.29433</td>
<td>11</td>
<td>16</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>13</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>12</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>12</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>6022</td>
<td>11.312</td>
<td>83</td>
<td>83</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>5104828</td>
<td>11.488</td>
<td>83</td>
<td>83</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function</th>
<th>Latencies</th>
<th>Costs</th>
<th>Duration</th>
<th>Throughput</th>
<th>GC</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>6022</td>
<td>10.488</td>
<td>83</td>
<td>83</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>5104828</td>
<td>11.058</td>
<td>83</td>
<td>83</td>
<td></td>
</tr>
</tbody>
</table>

stackPush | Latencies | Costs | Duration | Throughput | GC   |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>6022</td>
<td>42.9456</td>
<td>11.312</td>
<td>83</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>5104828</td>
<td>39.8329</td>
<td>11.488</td>
<td>83</td>
<td></td>
</tr>
</tbody>
</table>

stackPop | Latencies | Costs | Duration | Throughput | GC   |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>6022</td>
<td>37.7567</td>
<td>10.488</td>
<td>83</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>5104828</td>
<td>34.3733</td>
<td>11.058</td>
<td>83</td>
<td></td>
</tr>
</tbody>
</table>
Latencies: |  |  |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>6038</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5103484</td>
</tr>
</tbody>
</table>

Costs: |  |  |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>6038</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5103484</td>
</tr>
</tbody>
</table>

ScamBalance/Cycles | 10016 | 0.556026 | 0.467851 | 0.00034382 | 1.56666 |

Heap Utilization | 2 | 0.00106049 | 0.000852378 | 0.0000457764 | 0.00166321 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 18799

total cycles required for GC: 1362619

bus utilization: 35.366%

utilization due to cache invalidation requests: 2.026%

icache hit rate (497898611 hits / 498006688 fetches): 99.978%

dcache hit rate (19496732 hits / 75146188 fetches): 25.945%

Simulation completed on Sun Jul 5 06:31:17 1992
CPU usage: 788.9 system, 117383.6 user

3.15 gc/lisp/prune/fastcpu/lowmem
dlxsimcg release:

Command line: fast/dlxsimcg 40000 lisp __start

This simulation begun on Thu Jul 2 11:16:56 1992

Statistics:
  Program image ends at 0x2f6f0

  total machine instructions executed: 524404669
  cycles stalled for instruction fetch: 113927125
  cycles stalled for memory operations: 52985556
  cycles stalled following loads: 485307327
  cycles stalled for floating point results: 0
  cycles stalled for floating point processors: 0
  cycles stalled for branch-delay instruction fetches: 140587
  cycles stalled for trap interfacing: 3254

  total machine cycles executed: 1653738918

  total number of traps executed: 84

Arbiter | Operation Invocations
---------|---------------------
Operations | number | mean* | std dev | range
---------|-------|-------|--------|------
copyBlock |       |       |        |      |
Latencies: | During GC | 0 | 0 | 0 | |
While GC idle | 0 | 0 | 0 | |
Costs: | During GC | 0 | 0 | 0 | 0 | 0 |
<table>
<thead>
<tr>
<th>Method</th>
<th>GC Idle</th>
<th>AllocInitRec</th>
<th>AllocRec</th>
<th>AllocDSlice</th>
<th>AllocDSSubSlice</th>
<th>AllocTSlice</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>tendingDone</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>12</td>
<td>-1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>12</td>
<td>-1</td>
<td>0</td>
<td>5</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocInitRec</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>176</td>
<td>77.9261</td>
<td>8.90074</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>18331</td>
<td>78.0041</td>
<td>10.177</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>176</td>
<td>70.0341</td>
<td>7.69705</td>
<td>37</td>
<td>79</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>18331</td>
<td>70.6901</td>
<td>8.46056</td>
<td>11</td>
<td>91</td>
<td></td>
</tr>
<tr>
<td><strong>copyPush</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>initBlock</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>321</td>
<td>34.4019</td>
<td>46.6212</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>321</td>
<td>31.2804</td>
<td>46.2537</td>
<td>15</td>
<td>213</td>
<td></td>
</tr>
<tr>
<td><strong>tendDesc</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>155</td>
<td>49.9806</td>
<td>20.9952</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>155</td>
<td>43.5419</td>
<td>21.7211</td>
<td>5</td>
<td>65</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocRec</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>301</td>
<td>34.3887</td>
<td>3.32642</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>301</td>
<td>22.3953</td>
<td>3.47081</td>
<td>21</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td><strong>allocDSSlice</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTSlice</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocDSSubSlice</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTSubSlice</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

AllocDStack
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

AllocTStack
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

StackPush
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

StackPop
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

ScanBalance/Cycles:

Heap Utilization:

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 18809
total cycles required for GC: 17283704

bus utilization: 45.51%
utilization due to cache invalidation requests: 3.744%
icache hit rate ( 635027458 hits / 635130464 fetches): 99.983%
dcache hit rate ( 19495615 hits / 102581046 fetches): 19.005%

Simulation completed on Sat Jul 4 21:08:13 1992
CPU usage: 851.3 system, 117110.6 user

3.16 gc/lisp/prune/fastcpu/highmem
dixsimc release:

Command line: fast/dixsimc .80000 lisp _start

This simulation begun on Mon Jun 29 11:49:30 1992
Statistics:
Program image ends at 0x2f6f0

<table>
<thead>
<tr>
<th>total machine instructions executed:</th>
<th>510686749</th>
</tr>
</thead>
<tbody>
<tr>
<td>cycles stalled for instruction fetch:</td>
<td>110181396</td>
</tr>
<tr>
<td>cycles stalled for memory operations:</td>
<td>447017369</td>
</tr>
<tr>
<td>cycles stalled following loads:</td>
<td>445714489</td>
</tr>
<tr>
<td>cycles stalled for floating point results:</td>
<td>0</td>
</tr>
<tr>
<td>cycles stalled for floating point processors:</td>
<td>0</td>
</tr>
<tr>
<td>cycles stalled for branch-delay instruction fetches:</td>
<td>132777</td>
</tr>
<tr>
<td>cycles stalled for trap interfacing:</td>
<td>3254</td>
</tr>
<tr>
<td>total machine cycles executed:</td>
<td>1513736034</td>
</tr>
<tr>
<td>total number of traps executed:</td>
<td>84</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendingDone</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendDesc</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>301</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>301</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>10277</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5100583</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>10277</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5100583</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>10330</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5099202</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>10330</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5099202</td>
</tr>
<tr>
<td>ScanBalance/Cycles</td>
<td>15290</td>
</tr>
<tr>
<td>Heap Utilization</td>
<td>2</td>
</tr>
</tbody>
</table>

49
3.17 gc/troff/osmpaper/slowcpu/lowmem

dlxsimgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: dlxsimgc.200000 troff `_start osmpaper

This simulation begun on Mon Jun 29 14:23:38 1992

Statistics:
  Program image ends at 0x17d150

  total machine instructions executed: 612749907
  cycles stalled for instruction fetch: 259989925
  cycles stalled for memory operations: 501265074
  cycles stalled following loads: 348560099
  cycles stalled for floating point results: 62315
  cycles stalled for floating point processors: 0
  cycles stalled for branch-delay instruction fetches: 16350951
  cycles stalled for trap interfacing: 6195725

  total machine cycles executed: 1745173996

  total number of traps executed: 176035

Arbiter | Operation Invocations
---------|------------------|------------------|------------------|------------------|------------------|

<table>
<thead>
<tr>
<th>Operations</th>
<th>number</th>
<th>mean*</th>
<th>std dev</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td>copyBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>72</td>
<td>136.5</td>
<td>5.42296</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>1843</td>
<td>113.06</td>
<td>31.1127</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>72</td>
<td>130.444</td>
<td>4.82041</td>
<td>122</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1843</td>
<td>110.664</td>
<td>33.1889</td>
<td>30</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
<td>-1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
<td>-1</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>186</td>
<td>34.586</td>
<td>4.29653</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>35719</td>
<td>34.6205</td>
<td>10.5874</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>186</td>
<td>21.6129</td>
<td>5.33227</td>
<td>19</td>
</tr>
<tr>
<td>While GC idle</td>
<td>35719</td>
<td>27.1704</td>
<td>9.93991</td>
<td>5</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>175</td>
<td>128.817</td>
<td>16.3254</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>6138</td>
<td>116.655</td>
<td>38.5222</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>175</td>
<td>123.829</td>
<td>15.8595</td>
<td>34</td>
</tr>
<tr>
<td>While GC idle</td>
<td>6138</td>
<td>111.848</td>
<td>38.0286</td>
<td>8</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>21</td>
<td>39.7143</td>
<td>10.7571</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>21</td>
<td>22.9048</td>
<td>10.2221</td>
<td>3</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>8</td>
<td>19.25</td>
<td>0.707107</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>748</td>
<td>27.6631</td>
<td>14.5389</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>8</td>
<td>16</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>While GC idle</td>
<td>748</td>
<td>14.1203</td>
<td>2.4416</td>
<td>5</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
While GC idle: 1 | 13 | 0 | 
Costs: 
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 1 | 12 | 0 | 12 | 12 |

allocStackLatencies: 
| During GC | 0 | 0 | 0 | 
| While GC idle | 0 | 0 | 0 | 
Costs: 
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

stackPushLatencies: 
| During GC | 34769 | 68.2621 | 36.5706 | 
| While GC idle | 5759472 | 59.0112 | 31.6456 | 
Costs: 
| During GC | 34769 | 62.9631 | 36.7512 | 15 | 142 |
| While GC idle | 5759472 | 53.6482 | 31.6833 | 15 | 136 |

stackPopLatencies: 
| During GC | 27664 | 26.3408 | 5.59557 | 
| While GC idle | 5138749 | 23.1 | 5.7713 | 
Costs: 
| During GC | 27664 | 21.5266 | 4.52645 | 9 | 28 |
| While GC idle | 5138749 | 19.013 | 4.65665 | 9 | 28 |

ScanBalance/Cycles: 76134 | 0.346188 | 3.76074 | 0.000130174 | 871.833 
Heap Utilization: 2 | 0.0121918 | 0.0151261 | 0.00149918 | 0.0228844 

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 36662 
total cycles required for GC: 11484009 
bus utilization: 44.362%
utilization due to cache invalidation requests: 3.054%
icache hit rate (70360950 hits / 722770716 fetches): 97.355%
dcache hit rate (40709858 hits / 124976974 fetches): 32.574%

Simulation completed on Fri Jul 3 11:40:05 1992 
CPU usage: 204.5 system, 74482.4 user 

3.18 gc/troff/osmpaper/slowcpu/highmem 
dlxsimgc release: 
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $ 
Command line: /usr/schmidt/gnu/sim/slowcpu.morestats/dlxsimgc.400000 
troff --start ..../data/osmpaper 
This simulation begun on Sat Jun 27 08:34:19 1992 
Statistics: 
Program image starts at 0x17d150 
total machine instructions executed: 589546754 
cycles stalled for instruction fetch: 229992731 
cycles stalled for memory operations: 334293750 
cycles stalled following loads: 282859617 
cycles stalled for floating point results: 62275
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 14419073
cycles stalled for trap interfacing: 3862021

---------
total machine cycles executed: 1455036221
---------
total number of traps executed: 176035

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Operations</td>
</tr>
<tr>
<td></td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1915</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1915</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>35906</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>35906</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>6313</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>6313</td>
</tr>
<tr>
<td>tendingDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>755</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>755</td>
</tr>
<tr>
<td>allocDSSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
</tbody>
</table>

53
During GC | 0 | 0 | 0 | 0 | 0 | 0 |
While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |
Costs: 
During GC | 0 | 0 | 0 | 0 | 0 | 0 |
While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |
allocTSlice 
Latencies: 
During GC | 0 | 0 | 0 | 0 | 0 | 0 |
While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |
Costs: 
During GC | 0 | 0 | 0 | 0 | 0 | 0 |
While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |
allocDSSubSlice 
Latencies: 
During GC | 0 | 0 | 0 | 0 | 0 | 0 |
While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |
allocDStack 
Latencies: 
During GC | 0 | 0 | 0 | 0 | 0 | 0 |
While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |
stackPush 
Latencies: 
During GC | 0 | 0 | 0 | 0 | 0 | 0 |
While GC idle | 5795059 | 54.8673 | 32.7391 | 0 | 0 | 0 |
Costs: 
During GC | 0 | 0 | 0 | 0 | 0 | 0 |
While GC idle | 5795059 | 49.726 | 32.9963 | 15 | 131 |
stackPop 
Latencies: 
During GC | 0 | 0 | 0 | 0 | 0 | 0 |
While GC idle | 5167221 | 17.7133 | 3.5333 | 0 | 0 | 0 |
Costs: 
During GC | 0 | 0 | 0 | 0 | 0 | 0 |
While GC idle | 5167221 | 14.5147 | 4.18818 | 9 | 23 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x88d068)

number of allocations unimpeded by GC: 36662
total cycles required for GC: 0
bus utilization: 36.482%
utilization due to cache invalidation requests: 3.662%
icache hit rate (67468612 hits / 693769216 fetches): 97.244%
dcache hit rate (40666609 hits / 119173114 fetches): 34.124%

Simulation completed on Wed Jul 1 08:37:11 1992
CPU usage: 1713.5 system, 106998.7 user

3.19  gc/troff/osmpaper/fastcpu/lowmem
dlxsimmgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $
Command line: fastcpu/dlxsimmgc.200000 troff ...start osmpaper
This simulation begun on Wed Jul 8 20:33:07 1992
Statistics:
Program image ends at 0x17d150

| total machine instructions executed: 741606808 |
| cycles stalled for instruction fetch: 367388938 |
| cycles stalled for memory operations: 864717237 |
| cycles stalled following loads: 671910823 |
| cycles stalled for floating point results: 61364 |
| cycles stalled for floating point processors: 0 |
| cycles stalled for branch-delay instruction fetches: 22864007 |
| cycles stalled for trap interfacing: 11162811 |
| total machine cycles executed: 2679711988 |
| total number of traps executed: 176035 |

<table>
<thead>
<tr>
<th>Arbiters</th>
<th>Operation Invocations</th>
<th>Operations</th>
<th>number</th>
<th>mean</th>
<th>std dev</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>163</td>
<td>269.601</td>
<td>5.18847</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>1752</td>
<td>220.784</td>
<td>65.166</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>163</td>
<td>262.448</td>
<td>7.39465</td>
<td>243</td>
<td>275</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>1752</td>
<td>218.309</td>
<td>67.4711</td>
<td>9</td>
<td>327</td>
<td></td>
</tr>
<tr>
<td>tendiendoDone</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
<td>-1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
<td>-1</td>
<td>0</td>
<td>5</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>allocInitRec</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>245</td>
<td>54.551</td>
<td>10.5963</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>35660</td>
<td>60.3344</td>
<td>19.6291</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>245</td>
<td>44.151</td>
<td>11.2351</td>
<td>27</td>
<td>127</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>35660</td>
<td>53.3469</td>
<td>19.8929</td>
<td>11</td>
<td>139</td>
<td></td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

55
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
<th>Costs:</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>304</td>
<td>220.296</td>
<td>24.0353</td>
</tr>
<tr>
<td>While GC idle</td>
<td>6009</td>
<td>185.894</td>
<td>60.6673</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>304</td>
<td>214.539</td>
<td>23.0247</td>
</tr>
<tr>
<td>While GC idle</td>
<td>6009</td>
<td>181.568</td>
<td>61.4822</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>21</td>
<td>65.8571</td>
<td>17.9284</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>21</td>
<td>44.8095</td>
<td>20.4441</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>9</td>
<td>35</td>
<td>3</td>
</tr>
<tr>
<td>While GC idle</td>
<td>747</td>
<td>41.0991</td>
<td>11.8193</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>9</td>
<td>31</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>747</td>
<td>27.2383</td>
<td>4.87506</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>35</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>23</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
**Costs:**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**stackPush**

**Latencies:**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>64353</td>
<td>127.738</td>
<td>72.2539</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>5729941</td>
<td>109.704</td>
<td>60.7081</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Costs:**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>64353</td>
<td>122.596</td>
<td>71.4394</td>
<td>29</td>
<td>283</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5729941</td>
<td>104.747</td>
<td>59.9529</td>
<td>29</td>
<td>271</td>
</tr>
</tbody>
</table>

**stackPop**

**Latencies:**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>51029</td>
<td>46.4834</td>
<td>9.38738</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>5115402</td>
<td>42.4637</td>
<td>9.08892</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Costs:**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>51029</td>
<td>41.2442</td>
<td>9.22944</td>
<td>17</td>
<td>55</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5115402</td>
<td>37.023</td>
<td>9.3153</td>
<td>17</td>
<td>55</td>
</tr>
</tbody>
</table>

**ScanBalance/Cycles:**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>97106</td>
<td>0.238548</td>
<td>3.07352</td>
<td>4.44109e-05</td>
<td>872.833</td>
<td></td>
</tr>
</tbody>
</table>

**Heap Utilization:**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.0205421</td>
<td>0.025458</td>
<td>10.0254659</td>
<td>0.0385437</td>
<td></td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

---

**number of allocations unimpeded by GC:** 36662

**total cycles required for GC:** 32906976

**bus utilization:** 53.014%

**utilization due to cache invalidation requests:** 5.967%

**icache hit rate (864658957 hits / 883799919 fetches):** 97.836%

**dcache hit rate (40709060 hits / 157189906 fetches):** 25.896%

Simulation completed on Fri Jul 10 06:40:09 1992

CPU usage: 578.8 system, 105665.1 user

### 3.20 gc/troff/osmpaper/fastcpu/highmem

dlxsimc release:

$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp$

Command line: fastcpu/dlxsimc.400000 troff __start osmpaper

This simulation begun on Fri Jul 3 13:31:41 1992

Statistics:

Program image ends at 0x17d150

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>total machine instructions executed:</td>
<td>700384797</td>
</tr>
<tr>
<td>cycles stalled for instruction fetch:</td>
<td>309741270</td>
</tr>
<tr>
<td>cycles stalled for memory operations:</td>
<td>522774050</td>
</tr>
<tr>
<td>cycles stalled following loads:</td>
<td>545251345</td>
</tr>
<tr>
<td>cycles stalled for floating point results:</td>
<td>61313</td>
</tr>
<tr>
<td>cycles stalled for floating point processors:</td>
<td>0</td>
</tr>
<tr>
<td>cycles stalled for branch-delay instruction fetches:</td>
<td>19109686</td>
</tr>
<tr>
<td>cycles stalled for trap interfacing:</td>
<td>6495290</td>
</tr>
</tbody>
</table>

total machine cycles executed: 1203817751

total number of traps executed: 176035
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Operations</td>
</tr>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>tendingDone</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>allocInitRec</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>copyPush</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>initBlock</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>tendDesc</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>allocRec</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>allocDSlice</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>allocTSlice</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
</tbody>
</table>

58
<table>
<thead>
<tr>
<th>Costs:</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>35</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>23</td>
<td>0</td>
<td>23</td>
<td>23</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>5794241</td>
<td>101.714</td>
<td>61.262</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5794241</td>
<td>96.0458</td>
<td>60.4443</td>
<td>29</td>
<td>261</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>5166413</td>
<td>33.8098</td>
<td>7.84367</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5166413</td>
<td>28.0304</td>
<td>8.37669</td>
<td>17</td>
<td>45</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x88d0b8)

d number of allocations unimpeded by GC: 36660
total cycles required for GC: 0

bus utilization: 44.44%utilization due to cache invalidation requests: 7.598%

icache hit rate ( 813151793 hits / 832271327 fetches): 97.703%
dcache hit rate ( 40663188 hits / 146885717 fetches): 27.684%

Simulation completed on Tue Jul 7 16:12:18 1992
CPU usage: 202.2 system, 91747.0 user
3.21 gc/troff/toplas/slowcpu/lowmem

dlxsimgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: dlxsimgc.200000 troff _-start -mps -ms -mpsfig toplas

This simulation begun on Mon Jun 29 14:22:49 1992

Statistics:
Program image ends at 0x17d160

<table>
<thead>
<tr>
<th></th>
<th>total machine instructions executed: 1275720487</th>
</tr>
</thead>
<tbody>
<tr>
<td>cycles stalled for instruction fetch: 62849652</td>
<td></td>
</tr>
<tr>
<td>cycles stalled for memory operations: 1047023897</td>
<td></td>
</tr>
<tr>
<td>cycles stalled following loads: 727355643</td>
<td></td>
</tr>
<tr>
<td>cycles stalled for floating point results: 106461</td>
<td></td>
</tr>
<tr>
<td>cycles stalled for floating point processors: 0</td>
<td></td>
</tr>
<tr>
<td>cycles stalled for branch-delay instruction fetches: 46444229</td>
<td></td>
</tr>
<tr>
<td>cycles stalled for trap interfacing: 11055268</td>
<td></td>
</tr>
</tbody>
</table>
| total machine cycles executed: ---
| total number of traps executed: 3736205637

<table>
<thead>
<tr>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
</tr>
<tr>
<td>arbiter</td>
</tr>
<tr>
<td>copyBlock</td>
</tr>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>tendingDone</td>
</tr>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>allocInitRec</td>
</tr>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
</tr>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
</tr>
<tr>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>----------------------</td>
</tr>
<tr>
<td><strong>tendDesc</strong></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>allocRec</strong></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>allocDSlice</strong></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>allocDSSubSlice</strong></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>allocDStack</strong></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>allocTStack</strong></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>stackPush</strong></td>
</tr>
<tr>
<td><strong>Latencies:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
</tr>
<tr>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
</tr>
</tbody>
</table>
Latencies:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>116799</td>
<td>26.1938</td>
<td>5.54689</td>
</tr>
<tr>
<td>While GC idle</td>
<td>10405007</td>
<td>24.0148</td>
<td>5.70026</td>
</tr>
</tbody>
</table>

Costs:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>
| During GC        | 116799 | 21.4045| 4.47164| 9| 28
| While GC idle    | 10405007| 19.7948| 4.46824| 9| 28

ScamBalance/Cycles:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>
|                 | 283108| 0.329755| 3.4504616.44423e-05| 871.833

Heap Utilization:

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>7</td>
<td>0.008279124</td>
<td>0.0107417</td>
<td>0.00206375</td>
<td>0.0274467</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 69303
total cycles required for GC: 46563514

bus utilization: 45.32%
utilization due to cache invalidation requests: 3.028%
icache hit rate (1453898898 hits / 1506098432 fetches): 96.534%
dcache hit rate ( 86463355 hits / 261825584 fetches): 33.025%

Simulation completed on Tue Jul 7 07:51:19 1992
CPU usage: 388.5 system, 156900.7 user

3.22 gc/troff/toplas/slowcpu/highmem
dlxsimgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: dlxsimgc.400000 troff __start -mps -ms -msfig toplas

This simulation begun on Mon Jun 29 14:23:13 1992

Statistics:

Program image ends at 0x17d160
total machine instructions executed: 1251369971
cycles stalled for instruction fetch: 592435447
cycles stalled for memory operations: 872094880
cycles stalled following loads: 656954889
cycles stalled for floating point results: 106279
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 43693261
cycles stalled for trap interfacing: 8926944

---------
total machine cycles executed: 3425581771

---------
total number of traps executed: 300905

---------------------------------------------------------------------
Arbiter | Operation Invocations
---------------------------------------------------------------------
Operations | number | mean* | std dev | range
---------------------------------------------------------------------
copyBlock |        |        |        |
Latencies: |        |        |        |
| During GC | 0 | 0 | 0 |
| While GC idle | 2165 | 113.365 | 30.2682 |
Costs: |        |        |        |
| During GC | 0 | 0 | 0 | 0

62
<table>
<thead>
<tr>
<th>Method</th>
<th>While GC idle</th>
<th>111.425</th>
<th>32.2552</th>
<th>25</th>
<th>173</th>
</tr>
</thead>
<tbody>
<tr>
<td>tendingDone</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>115</td>
<td>37.6348</td>
<td>8.19213</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>67035</td>
<td>35.5204</td>
<td>13.3851</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>115</td>
<td>27.0696</td>
<td>8.64882</td>
<td>19</td>
<td>64</td>
</tr>
<tr>
<td>While GC idle</td>
<td>67035</td>
<td>26.9743</td>
<td>12.3371</td>
<td>5</td>
<td>70</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>10580</td>
<td>105.234</td>
<td>48.7432</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>10580</td>
<td>100.438</td>
<td>48.3834</td>
<td>8</td>
<td>142</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>11</td>
<td>41.2727</td>
<td>9.04534</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>11</td>
<td>22.6364</td>
<td>9.38374</td>
<td>3</td>
<td>32</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>2146</td>
<td>26.3085</td>
<td>14.4748</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2146</td>
<td>12.9734</td>
<td>2.44449</td>
<td>11</td>
<td>16</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

63
Latencies:
  | | | | |
During GC | 0 | 0 | 0 | |
While GC idle | 0 | 0 | 0 |
Costs:
  | | | | |
During GC | 0 | 0 | 0 | 0 |
While GC idle | 0 | 0 | 0 |
allocDStack Latencies:
  | | | | |
During GC | 0 | 0 | 0 |
While GC idle | 1 | 13 |
Costs:
  | | | | |
During GC | 0 | 0 | 0 | 0 |
While GC idle | 1 | 12 | 0 |
allocTStack Latencies:
  | | | | |
During GC | 0 | 0 | 0 |
While GC idle | 0 | 0 |
Costs:
  | | | | |
During GC | 0 | 0 | 0 | 0 |
While GC idle | 0 | 0 |
stackPush Latencies:
  | | | | |
During GC | 23370 | 60.052 | 30.0879 |
While GC idle | 11915762 | 58.2571 | 33.0582 |
Costs:
  | | | | |
During GC | 23370 | 54.6948 | 30.1828 | 15 | 136 |
While GC idle | 11915762 | 53.0014 | 33.214 | 15 | 136 |
stackPop Latencies:
  | | | | |
During GC | 21133 | 25.9307 | 5.7767 |
While GC idle | 10500673 | 21.2074 | 5.7284 |
Costs:
  | | | | |
During GC | 21133 | 21.2041 | 4.47575 | 9 | 28 |
While GC idle | 10500673 | 17.4735 | 5.02844 | 9 | 28 |
ScanBalance/Cycles | 51065 | 0.456006 | 0.459012 | 0.000144269 | 11.1814 |
Heap Utilization | 1 | 0.00113297 | 0 | 0.00113297 | 0.00113297 |
----------------------------------------------------------------------------------
* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 69297
total cycles required for GC: 7525133

bus utilization: 41.806%
utilization due to cache invalidation requests: 3.303%
icache hit rate (1423474225 hits / 147563065 fetches): 96.463%
dcache hit rate ( 86437655 hits / 255738012 fetches): 33.799%

Simulation completed on Tue Jul 7 00:21:40 1992
CPU usage: 382.6 system, 149363.10 user

3.23 gc/troff/toplas/fastcpu/lowmem
dlxsimc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: fastcpu/dlxsimc.200000 troff __start -mps -ms -mpsfig toplas
This simulation begun on Tue Jun 30 19:36:54 1992
Statistics:
Program image ends at 0x71d160

<table>
<thead>
<tr>
<th>Statistics</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>total machine instructions executed</td>
<td>1547218941</td>
</tr>
<tr>
<td>cycles stalled for instruction fetch</td>
<td>89197106</td>
</tr>
<tr>
<td>cycles stalled for memory operations</td>
<td>1618029881</td>
</tr>
<tr>
<td>cycles stalled following loads</td>
<td>1412837597</td>
</tr>
<tr>
<td>cycles stalled for floating point results</td>
<td>105052</td>
</tr>
<tr>
<td>cycles stalled for floating point processors</td>
<td>0</td>
</tr>
<tr>
<td>cycles stalled for branch-delay instruction fetches</td>
<td>65456744</td>
</tr>
<tr>
<td>cycles stalled for trap interfacing</td>
<td>20017483</td>
</tr>
</tbody>
</table>

total machine cycles executed: 5755582804

total number of traps executed: 300905

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Operations</td>
</tr>
<tr>
<td></td>
<td>arbiter</td>
</tr>
<tr>
<td></td>
<td>copyBlock</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>tendDesc</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>allocInitRec</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>copyPush</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>initBlock</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>tendDesc</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>allocRec</td>
</tr>
</tbody>
</table>

65
<table>
<thead>
<tr>
<th>Latencies:</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>46</td>
<td>41.0217</td>
<td>11.5075</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>2102</td>
<td>42.941</td>
<td>12.7807</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>46</td>
<td>31</td>
<td>0</td>
<td>31</td>
<td>31</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2102</td>
<td>27.9029</td>
<td>4.64536</td>
<td>11</td>
<td>31</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>35</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>23</td>
<td>0</td>
<td>23</td>
<td>23</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>259515</td>
<td>121.215</td>
<td>68.1869</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>11679617</td>
<td>112.358</td>
<td>62.6213</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>259515</td>
<td>116.134</td>
<td>67.4085</td>
<td>29</td>
<td>295</td>
</tr>
<tr>
<td>While GC idle</td>
<td>11679617</td>
<td>107.501</td>
<td>61.7848</td>
<td>29</td>
<td>271</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>216579</td>
<td>46.5708</td>
<td>9.30618</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>10305227</td>
<td>44.0176</td>
<td>8.86463</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>216579</td>
<td>41.3226</td>
<td>9.11945</td>
<td>17</td>
<td>55</td>
</tr>
<tr>
<td>While GC idle</td>
<td>10305227</td>
<td>38.5858</td>
<td>8.93816</td>
<td>17</td>
<td>55</td>
</tr>
<tr>
<td>ScanBalance/Cycles</td>
<td>356512</td>
<td>0.226983</td>
<td>2.59936</td>
<td>2.49775e-05</td>
<td>1306.19</td>
</tr>
<tr>
<td>Heap Utilization</td>
<td>7</td>
<td>0.0153291</td>
<td>0.0167066</td>
<td>0.003479</td>
<td>0.0424614</td>
</tr>
</tbody>
</table>
* A value of -1 signifies that the mean is not available due
to unclaimed results.

number of allocations unimpeded by GC: 69303
        total cycles required for GC: 132294742
        bus utilization: 53.93%
        utilization due to cache invalidation requests: 5.90%
        icache hit rate (1793153360 hits / 1845253969 fetches): 97.17%
        dcache hit rate (86469293 hits / 329700195 fetches): 26.22%

Simulation completed on Wed Jul 8 19:15:49 1992
CPU usage: 586.1 system, 223769.3 user

3.24 gc/troff/toplas/fastcpu/highmem

dlxsimc release:
    $Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: /usr/schmidt/gnu/gc/sim/fastcpu/dlxsimc.400000 troff
               __start -mps -ms -mpsfig .../data/toplas

This simulation begun on Fri Jul 3 18:10:17 1992

Statistics:
        Program image ends at 0x17d170

        total machine instructions executed: 1501826171
        cycles stalled for instruction fetch: 819993657
        cycles stalled for memory operations: 1457937081
        cycles stalled following loads: 1273924928
        cycles stalled for floating point results: 94986
        cycles stalled for floating point processors: 0
        cycles stalled for branch-delay instruction fetches: 59791628
        cycles stalled for trap interfacing: 15697214

        total machine cycles executed: 5129265865

        total number of traps executed: 300912

---

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>-----------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
</tbody>
</table>

67
<table>
<thead>
<tr>
<th>allocInitRec</th>
<th>Latencies:</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>213</td>
<td>58.2254</td>
<td>12.8513</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>66926</td>
<td>59.8942</td>
<td>24.9252</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>213</td>
<td>48.6338</td>
<td>13.0383</td>
<td>37</td>
<td>95</td>
</tr>
<tr>
<td>While GC idle</td>
<td>66926</td>
<td>52.9157</td>
<td>24.7055</td>
<td>11</td>
<td>139</td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>10538</td>
<td>170.221</td>
<td>79.0403</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>10538</td>
<td>165.695</td>
<td>79.1369</td>
<td>15</td>
<td>249</td>
</tr>
<tr>
<td>tendDesc</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>12</td>
<td>64.6667</td>
<td>19.4625</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>12</td>
<td>41.3333</td>
<td>19.6854</td>
<td>5</td>
<td>63</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>2145</td>
<td>42.9147</td>
<td>12.7257</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2145</td>
<td>24.9487</td>
<td>4.88937</td>
<td>21</td>
<td>31</td>
</tr>
<tr>
<td>allocDSlice</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSStack</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
During GC | 0 | 0 | 0 | 0
| While GC idle | 1 | 35 | 0 | 0
| Costs:  
| During GC | 0 | 0 | 0 | 0 | 0 | 0
| While GC idle | 1 | 23 | 0 | 23 | 23 | 23
| allocTStack 
| Latencies:  
| During GC | 0 | 0 | 0 | 0 | 0 | 0
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0
| Costs:  
| During GC | 0 | 0 | 0 | 0 | 0 | 0
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0
| stackPush | 
| Latencies:  
| During GC | 42714 | 110.022 | 57.2453 | 1
| While GC idle | 11892189 | 108.216 | 63.018 | 1
| Costs:  
| During GC | 42714 | 105.016 | 56.6742 | 29 | 283
| While GC idle | 11892189 | 102.99 | 62.172 | 29 | 271
| stackPop | 
| Latencies:  
| During GC | 38950 | 45.6949 | 9.19091 | 1
| While GC idle | 10479145 | 39.4532 | 9.78162 | 1
| Costs:  
| During GC | 38950 | 40.4582 | 9.12291 | 17 | 55
| While GC idle | 10479145 | 33.8874 | 10.0624 | 17 | 55
| ScmBalance/Cycles | 72004 | 0.304966 | 2.44958 | 4.92587e-05 | 654.625
| Heap Utilization | 1.000190163 | 0.000190163 | 0.000190163

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 692985
total cycles required for GC: 20883990

bus utilization: 50.24%
utilization due to cache invalidation requests: 6.615%
icache hit rate (1736514377 hits / 1788707242 fetches): 97.082%
dcache hit rate (86323667 hits / 318368974 fetches): 27.114%

Simulation completed on Sat Jul 11 09:34:43 1992
CPU usage: 2291.10 system, 316385.9 user

3.25 nogc/sft/small/slowcpu
dlsximgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp$

Command line: dlxsimgc.40000 sftnogc ___start small

This simulation begun on Wed Jun 24 15:42:24 1992

Statistics:
Program image ends at 0x1f950

total machine instructions executed: 61071302
cycles stalled for instruction fetch: 2690687
cycles stalled for memory operations: 16924109
cycles stalled following loads: 17035730
cycles stalled for floating point results: 1457305

69
cycles stalled for floating point processors: 16598
cycles stalled for branch-delay instruction fetches: 4692
cycles stalled for trap interfacing: 3760

----------------------
total machine cycles executed: 99204183
----------------------
total number of traps executed: 207

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>------------------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

70
<table>
<thead>
<tr>
<th>Method</th>
<th>During GC</th>
<th>While GC idle</th>
<th>Costs</th>
</tr>
</thead>
<tbody>
<tr>
<td>allocTSlice</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPop</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x8200000)

number of allocations unimpeded by GC: 0
total cycles required for GC: 0

71
CPU usage: 304.7 system, 4988.7 user

3.26 nogc/sft/small/fastcpu

dlxsimgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp$

Command line: /usr/schmidt/gnu/gc/sim/fastcpu/dlxsimgc.40000 sft
              __start__.data/small

This simulation begun on Thu Jun 25 15:13:25 1992

Statistics:
  Program image ends at 0x1f950

    total machine instructions executed: 61071302
    cycles stalled for instruction fetch: 2753989
    cycles stalled for memory operations: 34089449
    cycles stalled following loads: 23864885
    cycles stalled for floating point results: 1457282
    cycles stalled for floating point processors: 16598
         cycles stalled for branch-delay instruction fetches: 9102
    cycles stalled for trap interfacing: 6542

    total machine cycles executed: 123269149

    total number of traps executed: 207

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>--------------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
</tbody>
</table>

| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

| copyPush | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |

| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

| initBlock | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |

| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

| tendDesc | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |

| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

| allocRec | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |

| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

| allocDSlice | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |

| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

| allocTSlice | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |

| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

| allocDSubSlice | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |

| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

| allocDSubSlice | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |

| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

| allocDStack | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | 0 |

73
| While GC idle | 0 | 0 | 0 |
| Costs: |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |
| allocTStack |
| Latencies: |
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 |
| Costs: |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |
| stackPush |
| Latencies: |
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 |
| Costs: |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |
| stackPop |
| Latencies: |
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 |
| Costs: |
| During GC | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x8200000)

number of allocations unimpeded by GC: 0
number of allocations impeded by GC: 0
total cycles required for GC: 0
bus utilization: 44.271%
utilization due to cache invalidation requests: 0.000%
icache hit rate ( 63679752 hits / 63685978 fetches): 99.990%
dcache hit rate ( 21490727 hits / 21558670 fetches): 99.685%

number of executions of ‘malloc’: 1
total cycles for ‘malloc’ executions: 10942
mean cycles per ‘malloc’ execution: 10942.000
standard deviation of ‘malloc’ execution times: 0.000
range of malloc costs: 10942 to 10942

Simulation completed on Fri Jun 26 00:40:28 1992
CPU usage: 38.9 system, 9008.7 user

3.27 nogc/sfft/medium/slowcpu

dlxsimgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: /usr/schmidt/gnmgc/sim/slowcpu.moresstats/dlxsimgc.40000
sfft __start ../data/medium

This simulation begun on Wed Jun 24 22:19:33 1992
Statistics:
Program image ends at 0x1f950

    total machine instructions executed: 243380460
    cycles stalled for instruction fetch: 10486935
    cycles stalled for memory operations: 67320254
    cycles stalled following loads: 67933120
    cycles stalled for floating point results: 5773368
    cycles stalled for floating point processors: 65748
    cycles stalled for branch-delay instruction fetches: 4966
    cycles stalled for trap interfacing: 8016

    total machine cycles executed: 394972868
    total number of traps executed: 431

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocDSlice</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTSlice</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocDSSubSlice</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTSubSlice</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocDStack</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTStack</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>stackPush</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>stackPop</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.
warning: no garbage collection activity (New = 0x820000)

number of allocations unimpeded by GC: 0
total cycles required for GC: 0

bus utilization: 27.646%
utilization due to cache invalidation requests: 0.000%
icache hit rate ( 253769918 hits / 253776477 fetches): 99.997%
dcache hit rate ( 85749463 hits / 86017654 fetches): 99.688%

number of executions of 'malloc': 1
total cycles for 'malloc' executions: 6602
mean cycles per 'malloc' execution: 6602.000
standard deviation of 'malloc' execution times: 0.000
range of malloc costs: 6602 to 6602

Simulation completed on Fri Jun 26 00:10:34 1992
CPU usage: 472.7 system, 32422.10 user

3.28 nogc/sfft/medium/fastcpu
dlxsimgc release:
 $Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp$

Command line: /usr/schmidt/gnu/c/sim/fastcpu/dlxsimgc.40000 sfft
__start __data/medium

This simulation begun on Fri Jun 26 07:44:09 1992

Statistics:
  Program image ends at 0x1f950

  total machine instructions executed: 243380460
  cycles stalled for instruction fetch: 10572861
  cycles stalled for memory operations: 135586730
  cycles stalled for floating point results: 5773342
  cycles stalled for floating point processors: 65749
  cycles stalled for branch-delay instruction fetches: 9602
  cycles stalled for trap interfacing: 13934

  total machine cycles executed: 490578649
  total number of traps executed: 431

-----------------------------------------------
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
</tbody>
</table>
Latencies:

<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>allocRec</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSlice</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Costs:

<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>allocRec</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSlice</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

78
<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

**warning:** no garbage collection activity (New = 0x8200000)

- number of allocations unimpeded by GC: 0
- total cycles required for GC: 0
- bus utilization: 44.20\%
- utilization due to cache invalidation requests: 0.000%
- icache hit rate ( 253769918 hits / 253776477 fetches): 99.997%
- dcache hit rate ( 85749463 hits / 86017654 fetches): 99.688%

- number of executions of 'malloc': 1
- total cycles for 'malloc' executions: 10942
- mean cycles per 'malloc' execution: 10942.000
- standard deviation of 'malloc' execution times: 0.000
- range of malloc costs: 10942 to 10942

Simulation completed on Sat Jun 27 11:58:34 1992
CPU usage: 273.3 system, 35546.7 user

### 3.29 nogc/lisp/db/slowcpu

dlxsimc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $
Command line: dlxsimgc.40000 lispnode __start

This simulation begun on Thu Jun 25 07:33:42 1992

Statistics:
  Program image ends at 0x14d40

  total machine instructions executed: 216417830
  cycles stalled for instruction fetch: 41155102
  cycles stalled for memory operations: 61266110
  cycles stalled following loads: 18834098
  cycles stalled for floating point results: 0
  cycles stalled for floating point processors: 0
  cycles stalled for branch-delay instruction fetches: 126851
  cycles stalled for trap interfacing: 19865

  total machine cycles executed: 337819856
  total number of traps executed: 1853

---------------------------------------------------------------------
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>----------------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
</tr>
<tr>
<td>---------------------</td>
<td>---</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocRec</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocDSlice</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTSlice</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTSubSlice</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocDStack</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTStack</strong></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
</tbody>
</table>
During GC| 0| 0| 0| 0| 0| 0  
While GC idle| 0| 0| 0| 0| 0| 0  

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x820000)

number of allocations unimpeded by GC: 0  
total cycles required for GC: 0  
bus utilization: 33.406%  
utilization due to cache invalidation requests: 0.000%  
icache hit rate (241408593 hits / 241503780 fetches): 99.961%  
dcache hit rate (49779741 hits / 50081169 fetches): 99.398%  
number of executions of 'malloc': 28933  
total cycles for 'malloc' executions: 7711619  
mean cycles per 'malloc' execution: 266.534  
standard deviation of 'malloc' execution times: 300.672  
range of malloc costs: 206 to 7813

Simulation completed on Fri Jun 26 05:29:34 1992  
CPU usage: 65.2 system, 16450.2 user

3.30 nogc/lisp/db/fastcpu

dlxsimc release:  
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $  

Command line: fastcpu/dlxsimc.40000 lispnogc __start

This simulation begun on Mon Jun 29 14:24:40 1992

Statistics:
  Program image ends at 0x14d40
  total machine instructions executed: 216417830  
cycles stalled for instruction fetch: 54136101  
cycles stalled for memory operations: 157121343  
cycles stalled following loads: 24564998  
cycles stalled for floating point results: 0  
cycles stalled for floating point processors: 0  
cycles stalled for branch-delay instruction fetches: 221154  
cycles stalled for trap interfacing: 41017  
  total machine cycles executed: 452502443
  total number of traps executed: 1853

--------------------------------------------------------------------------
Arbiter | Operation Invocations
--------------------------------------------------------------------------
Operations| number | mean* | std dev | range
--------------------------------------------------------------------------
copyBlock| | | | | |
Latencies:| | | | | |
  During GC| 0| 0| 0| 0| |

82
<table>
<thead>
<tr>
<th>Method</th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>allocInitRec</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSlice</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

| Latencies:   |
|--------------|-----------|---------------|
| During GC    | 0         | 0             |
| While GC idle| 0         | 0             |
| Costs:       |
| During GC    | 0         | 0             |
| While GC idle| 0         | 0             |

While GC idle | 0 | 0 | 0 | |
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
<th>Latencies:</th>
<th>Costs:</th>
<th>Latencies:</th>
<th>Costs:</th>
<th>Latencies:</th>
<th>Costs:</th>
<th>Latencies:</th>
<th>Costs:</th>
</tr>
</thead>
<tbody>
<tr>
<td>allocTSubSlice</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x820000)

duration of GCs: 0

total cycles for GC: 0

bus utilization: 49.324%
utilization due to cache invalidation requests: 0.000%
icache hit rate (241408593 hits / 241503780 fetches): 99.861%
dcache hit rate (49778330 hits / 50081169 fetches): 99.396%

number of executions of 'malloc': 28933

total cycles for 'malloc' executions: 11228357

mean cycles per 'malloc' execution: 388.081

standard deviation of 'malloc' execution times: 477.146

range of malloc costs: 310 to 12970

Simulation completed on Tue Jun 30 18:18:34 1992
CPU usage: 76.6 system, 19777.5 user
3.31 nogc/lisp/prune/slowcpu

dlxsimgc release:
   $Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp$

Command line: dlxsimgc.40000 lispnogc __start

This simulation begun on Wed Jun 24 15:43:01 1992

Statistics:
   Program image ends at 0x14d40

   total machine instructions executed: 118421396
   cycles stalled for instruction fetch: 22349583
   cycles stalled for memory operations: 34291053
   cycles stalled following loads: 10586520
   cycles stalled for floating point results: 0
   cycles stalled for floating point processors: 0
   cycles stalled for branch-delay instruction fetches: 94592
   cycles stalled for trap interfacing: 2771

   total machine cycles executed: 185745915
   total number of traps executed: 676

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>number</td>
</tr>
<tr>
<td>copyBlock Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>copyPush Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>allocInitRec Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>initBlock Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>tendingDone Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

85
<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Latencies:

| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 |

Costs:

| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x820000)

| number of allocations unimpeded by GC: | 0 |
| total cycles required for GC: | 0 |
| bus utilization: | 33.60% |
| utilization due to cache invalidation requests: | 0.000% |
| icache hit rate ( 131985860 hits / 132038195 fetches): | 99.945% |
| dcache hit rate ( 27158440 hits / 27333214 fetches): | 99.361% |

| number of executions of 'malloc': | 18795 |
| total cycles for 'malloc' executions: | 5061208 |
| mean cycles per 'malloc' execution: | 269.285 |
| standard deviation of 'malloc' execution times: | 307.073 |
| range of malloc costs: | 206 to 7913 |

Simulation completed on Thu Jun 25 03:55:50 1992
CPU usage: 310.6 system, 9084.3 user

3.32 nogc/lisp/prune/fastcpu

dlxsimc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: /usr/schmidt/gnucl/sim/fastcpu/dlxsimc.40000 lisp __start

This simulation begun on Fri Jun 26 07:45:23 1992

Statistics:
Program image ends at 0x14d40

| total machine instructions executed: | 118421396 |
| cycles stalled for instruction fetch: | 29555927 |
| cycles stalled for memory operations: | 86594619 |
| cycles stalled following loads: | 13962603 |
| cycles stalled for floating point results: | 0 |
| cycles stalled for floating point processors: | 0 |
| cycles stalled for branch-delay instruction fetches: | 166040 |
| cycles stalled for trap interfacing: | 9209 |
| total machine cycles executed: | 248709794 |
| total number of traps executed: | 676 |

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
</tbody>
</table>

87
<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>copyBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x820000)

number of allocations unimpeded by GC: 0
total cycles required for GC: 0

bus utilization: 49.61%
utilization due to cache invalidation requests: 0.000%
icache hit rate (131965860 hits / 132038195 fetches): 99.945%
dcache hit rate (27157479 hits / 27333214 fetches): 99.357%

number of executions of 'malloc': 18795
total cycles for 'malloc' executions: 7383828
mean cycles per 'malloc' execution: 392.861
standard deviation of 'malloc' execution times: 489.428
range of malloc costs: 310 to 12970

Simulation completed on Sat Jun 27 00:26:07 1992
3.33 nogc/troff/osmpaper/slowcpu

dlxsimgc release:
   $Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: /usr/schmidt/gnuc/sim/slowcpu.morestats/dlxsimgc.40000
troff --start ..data/osmpaper

This simulation begun on Sat Jun 27 08:35:52 1992

Statistics:
   Program image ends at 0xa1900

   total machine instructions executed: 229416495
   cycles stalled for instruction fetch: 167219872
   cycles stalled for memory operations: 87825421
   cycles stalled following loads: 39314453
   cycles stalled for floating point results: 59513
   cycles stalled for floating point processors: 0
   cycles stalled for branch-delay instruction fetches: 13638897
   cycles stalled for trap interfacing: 4384009
   total machine cycles executed: 541858660
   total number of traps executed: 176545

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendingDone</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
</tbody>
</table>
3.34 nogc/troff/osmpaper/fastcpu

dlxsimgc release:
   $Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: fastcpu/dlxsimgc.40000 troffnogc __start osmpaper

This simulation begun on Sat Jun 27 15:54:41 1992

Statistics:
   Program image ends at 0xa1900

   total machine instructions executed: 229384407
   cycles stalled for instruction fetch: 285908619
   cycles stalled for memory operations: 175873442
   cycles stalled following loads: 56919652
   cycles stalled for floating point results: 59376
   cycles stalled for floating point processors: 0
   cycles stalled for branch-delay instruction fetches: 24023973

warning: no garbage collection activity (New = 0x8200000)

number of allocations unimpeded by GC: 0
total cycles required for GC: 0

bus utilization: 49.199%
utilization due to cache invalidation requests: 0.000%
icache hit rate (235638798 hits / 248244428 fetches): 94.922%
dcache hit rate (56487535 hits / 56726401 fetches): 99.579%

number of executions of 'malloc': 36652
total cycles for 'malloc' executions: 13867806
mean cycles per 'malloc' execution: 378.364
standard deviation of 'malloc' execution times: 275.993
range of malloc costs: 173 to 6795

number of executions of 'free': 22746
total cycles for 'free' executions: 2947388
mean cycles per 'free' execution: 129.578
standard deviation of 'free' execution times: 116.371
range of free costs: 93 to 923

Simulation completed on Sun Jun 28 14:34:35 1992
CPU usage: 284.5 system, 35461.1 user
cycles stalled for trap interfacing:  8070141
total machine cycles executed:    780239610
total number of traps executed:   176545

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
</tbody>
</table>
Costs:

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x820000)

number of allocations unimpeded by GC: 0
total cycles required for GC: 0

bus utilization: 65.749%
utilization due to cache invalidation requests: 0.000%

94
icache hit rate ( 235605281 hits / 248209152 fetches): 94.922%
dcache hit rate ( 56479643 hits / 56717242 fetches): 99.581%

number of executions of 'malloc': 36650
total cycles for 'malloc' executions: 20941704
mean cycles per 'malloc' execution: 571.397
standard deviation of 'malloc' execution times: 453.204
range of malloc costs: 222 to 11356

number of executions of 'free': 22746
total cycles for 'free' executions: 3931815
mean cycles per 'free' execution: 172.857
standard deviation of 'free' execution times: 203.267
range of free costs: 117 to 1580

Simulation completed on Mon Jun 29 06:55:37 1992
CPU usage: 95.1 system, 27129.1 user

3.35 nogc/troff/toplas/slowcpu
dlxsimgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: /usr/schmidt/gnu/sim/slowcpu.morestats/dlxsimgc.40000
troff _-start -mps -ms -mpsfig .../data/toplas

This simulation begun on Sat Jun 27 16:06:28 1992

Statistics:
Program image ends at 0xa1920

total machine instructions executed: 480417580
cycles stalled for instruction fetch: 454762100
cycles stalled for memory operations: 176975457
cycles stalled following loads: 81315467
cycles stalled for floating point results: 92399
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 39208206
cycles stalled for trap interfacing: 7469261

  total machine cycles executed: 1240240470

total number of traps executed: 301572

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>---------------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------</td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>allocInitRec</strong> Latencies:</td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>copyPush</strong> Latencies:</td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>initBlock</strong> Latencies:</td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>tendDesc</strong> Latencies:</td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>allocRec</strong> Latencies:</td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>allocDSlice</strong> Latencies:</td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTSSlice</strong> Latencies:</td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>allocTSubSlice</strong> Latencies:</td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>Costs:</strong></td>
<td></td>
</tr>
<tr>
<td><strong>During GC</strong></td>
<td>0</td>
</tr>
<tr>
<td><strong>While GC idle</strong></td>
<td>0</td>
</tr>
</tbody>
</table>
### allocDStack

| Latencies: | | | | | |
|-------------------|---|---|---|---|
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | |
| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0 | |

### allocTStack

| Latencies: | | | | | |
|-------------------|---|---|---|---|
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | |
| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0 | |

### stackPush

| Latencies: | | | | | |
|-------------------|---|---|---|---|
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | |
| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0 | |

### stackPop

| Latencies: | | | | | |
|-------------------|---|---|---|---|
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | |
| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0 | | |

* A value of -1 signifies that the mean is not available due to unclaimed results.

---

**warning:** no garbage collection activity (New = 0x8200000)

- number of allocations unimpeded by GC: 0
- total cycles required for GC: 0

- bus utilization: 52.026%
- utilization due to cache invalidation requests: 0.000%
- icache hit rate (484881667 hits / 520886840 fetches): 93.088%
- dcache hit rate (117983148 hits / 118733824 fetches): 99.368%

- number of executions of ‘malloc’: 69273
- total cycles for ‘malloc’ executions: 25288269
- mean cycles per ‘malloc’ execution: 365.052
- standard deviation of ‘malloc’ execution times: 288.179
- range of malloc costs: 173 to 6795

- number of executions of ‘free’: 54123
- total cycles for ‘free’ executions: 7927704
- mean cycles per ‘free’ execution: 146.476
- standard deviation of ‘free’ execution times: 139.039
- range of free costs: 93 to 943

Simulation completed on Tue Jun 30 21:53:24 1992
CPU usage: 1653.6 system, 76814.6 user

3.36 nogc/troff/toplas/fastcpu
dlxsimgc release:
This simulation begun on Fri Jul 3 18:08:42 1992

Statistics:

  Program image ends at 0xa1920

  total machine instructions executed: 480417576
  cycles stalled for instruction fetch: 778470731
  cycles stalled for memory operations: 354348881
  cycles stalled following loads: 119181812
  cycles stalled for floating point results: 92458
  cycles stalled for floating point processors: 0
  cycles stalled for branch-delay instruction fetches: 69434987
  cycles stalled for trap interfacing: 13746736

  total machine cycles executed: 1815693161

  total number of traps executed: 301572

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operations</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>number</td>
<td>mean*</td>
</tr>
</tbody>
</table>

  copyBlock
  Latencies:
    During GC | 0 | 0 | 0 |
    While GC idle | 0 | 0 | 0 |
  Costs:
    During GC | 0 | 0 | 0 | 0 | 0 |
    While GC idle | 0 | 0 | 0 | 0 |
  tendDesc
  Latencies:
    During GC | 0 | 0 | 0 |
    While GC idle | 0 | 0 | 0 |
  Costs:
    During GC | 0 | 0 | 0 | 0 | 0 |
    While GC idle | 0 | 0 | 0 | 0 |
  allocInitRec
  Latencies:
    During GC | 0 | 0 | 0 |
    While GC idle | 0 | 0 | 0 |
  Costs:
    During GC | 0 | 0 | 0 | 0 | 0 |
    While GC idle | 0 | 0 | 0 | 0 |
  copyPush
  Latencies:
    During GC | 0 | 0 | 0 |
    While GC idle | 0 | 0 | 0 |
  Costs:
    During GC | 0 | 0 | 0 | 0 | 0 |
    While GC idle | 0 | 0 | 0 | 0 |
  initBlock
  Latencies:
    During GC | 0 | 0 | 0 |
    While GC idle | 0 | 0 | 0 |
  Costs:
    During GC | 0 | 0 | 0 | 0 | 0 |
    While GC idle | 0 | 0 | 0 | 0 |
  tendDesc
  Latencies:
    During GC | 0 | 0 | 0 |
    While GC idle | 0 | 0 | 0 | 0
<table>
<thead>
<tr>
<th>Latencies:</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
While GC idle | 0 | 0 | 0 |
Costs: 
   During GC | 0 | 0 | 0 | 0 | 0 | 0 |
   While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x820000)

number of allocations unimpeded by GC: 0
total cycles required for GC: 0

bus utilization: 68.187%
utilization due to cache invalidation requests: 0.000%
icache hit rate (484881670 hits / 520886849 fetches): 93.086%
dcache hit rate (117983099 hits / 118733827 fetches): 99.366%

number of executions of 'malloc': 69273
total cycles for 'malloc' executions: 38022730
mean cycles per 'malloc' execution: 548.882
standard deviation of 'malloc' execution times: 481.936
range of malloc costs: 222 to 11356

number of executions of 'free': 54123
total cycles for 'free' executions: 10909671
mean cycles per 'free' execution: 201.572
standard deviation of 'free' execution times: 242.465
range of free costs: 117 to 1617

Simulation completed on Mon Jul 6 10:48:18 1992
CPU usage: 660.9 system, 92137.0 user

3.37 Partial invalidation: sfft

dlxsimgc release:

Command line: dlxsimgc.partinv sfft __start small

This simulation begun on Fri Jul 17 16:22:44 1992

Statistics:
   Program image ends at 0x35d30

   total machine instructions executed: 71833617
   cycles stalled for instruction fetch: 5276480
   cycles stalled for memory operations: 26486080
   cycles stalled following loads: 29759009
   cycles stalled for floating point results: 1457401
   cycles stalled for floating point processors: 16599
   cycles stalled for branch-delay instruction fetches: 4071
   cycles stalled for trap interfacing: 4647

   total machine cycles executed: 134837904
   total number of traps executed: 199

---------------------------------------------------------------------
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>--------------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>345</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>345</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>71</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>71</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2</td>
</tr>
<tr>
<td>allocDSSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
</tbody>
</table>
During GC | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0
allocDSSubSlice
Latencies:
During GC | 0 | 0 | 0
While GC idle | 0 | 0 | 0
Costs:
During GC | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0
allocTSubSlice
Latencies:
During GC | 0 | 0 | 0
While GC idle | 0 | 0 | 0
Costs:
During GC | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0
allocDStack
Latencies:
During GC | 0 | 0 | 0
While GC idle | 1 | 13 | 0
Costs:
During GC | 0 | 0 | 0 | 0 | 0
While GC idle | 1 | 12 | 0 | 12 | 12
allocTStack
Latencies:
During GC | 0 | 0 | 0
While GC idle | 0 | 0 | 0
Costs:
During GC | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0
stackPush
Latencies:
During GC | 0 | 0 | 0
While GC idle | 174746 | 69.0038 | 33.554
Costs:
During GC | 0 | 0 | 0 | 0 | 0
While GC idle | 174746 | 64.0372 | 33.084 | 15 | 123
stackPop
Latencies:
During GC | 0 | 0 | 0
While GC idle | 137972 | 16.1013 | 0.949459
Costs:
During GC | 0 | 0 | 0 | 0 | 0
While GC idle | 137972 | 13.5799 | 2.78538 | 9 | 23

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x82dd38)

number of allocations unimpeded by GC: 3
total cycles required for GC: 0

bus utilization: 33.58%
utilization due to cache invalidation requests: 0.002%
icache hit rate ( 77095714 hits / 77102342 fetches): 99.991%
of 6627 misses, average icache miss cost: 6.635
of 6627 Traditional misses, average icache miss cost: 6.635
dcache hit rate ( 21729982 hits / 23517421 fetches): 92.400%
of 1.78744e+06 misses, average dcache miss cost: 3.276
of 2103 GC misses, average dcache miss cost: 11.777
of 67470 Traditional misses, average dcache miss cost: 12.108
3.38 Partial invalidation: lisp

dlxsimgc release:

Command line: dlxsimgc.40000.partinv lisp __start

This simulation begun on Fri Jul 17 16:57:23 1992

Statistics:
Program image ends at 0x2f6f0

| total machine instructions executed: 420819029 |
| cycles stalled for instruction fetch: 114639505 |
| cycles stalled for memory operations: 160102989 |
| cycles stalled following loads: 171881987 |
| cycles stalled for floating point results: 0 |
| cycles stalled for floating point processors: 0 |
| cycles stalled for branch-delay instruction fetches: 104660 |
| cycles stalled for trap interfacing: 1946 |
| total machine cycles executed: 867550106 |
| total number of traps executed: 82 |

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>16</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>16</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>384</td>
</tr>
<tr>
<td>While GC idle</td>
<td>18126</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>384</td>
</tr>
<tr>
<td>While GC idle</td>
<td>18126</td>
</tr>
<tr>
<td>copyPush Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td>-----------</td>
<td>------------</td>
</tr>
<tr>
<td>During GC</td>
<td>20</td>
</tr>
<tr>
<td>While GC idle</td>
<td>357</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>20</td>
</tr>
<tr>
<td>While GC idle</td>
<td>357</td>
</tr>
<tr>
<td>tendDesc</td>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
<td>207</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>207</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
<td>20</td>
</tr>
<tr>
<td>While GC idle</td>
<td>282</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>20</td>
</tr>
<tr>
<td>While GC idle</td>
<td>282</td>
</tr>
<tr>
<td>allocDSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td>Latencies:</td>
</tr>
</tbody>
</table>
During GC | 47187 | 42.9584 | 11.4873 |
While GC idle | 5063663 | 40.9362 | 11.1929 |

### Costs:

| During GC | 47187 | 38.4234 | 10.8256 | 15 | 128 |
| While GC idle | 5063663 | 36.7244 | 10.5558 | 15 | 128 |

### stackPop

### Latencies:

| During GC | 47451 | 25.6335 | 5.7733 |
| While GC idle | 5062071 | 23.8634 | 6.15075 |

### ScanBalance/Cycles:

| 59171 | 0.435572 | 5.48677 [0.00341705] | 1283 |

---

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 18813

total cycles required for GC: 8621942

bus utilization: 26.87%

utilization due to cache invalidation requests: 0.013%

icache hit rate (505584381 hits / 50694221 fetches): 99.97%

of 109839 misses, average icache miss cost: 7.011

of 109839 Traditional misses, average icache miss cost: 7.011

dcache hit rate (35598628 hits / 76684441 fetches): 46.42%

of 4.10858e+07 misses, average dcache miss cost: 2.982

of 165270 GC misses, average dcache miss cost: 29.670

of 4215 Traditional misses, average dcache miss cost: 17.445

Simulation completed on Sat Jul 18 17:34:04 1992

CPU usage: 87.4 system, 43618.6 user

### 3.39 Partial invalidation: troff

dlxsimgc release:


Command line: /usr/schmidt/gmgc/sim/partinv/dlxsimgc.200000.partinv
troff __start__/data/osmpaper

This simulation begun on Fri Jul 17 17:06:10 1992

Statistics:

Program image ends at 0x17d150

total machine instructions executed: 601139688

cycles stalled for instruction fetch: 244225976

cycles stalled for memory operations: 261967367

cycles stalled following loads: 276539411

cycles stalled for floating point results: 62369

cycles stalled for floating point processors: 0

cycles stalled for branch-delay instruction fetches: 17009293

cycles stalled for trap interfacing: 6195739

--------------

total machine cycles executed: 1407139843

total number of traps executed: 176035
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>90</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1825</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>90</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1825</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>198</td>
</tr>
<tr>
<td>While GC idle</td>
<td>35709</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>198</td>
</tr>
<tr>
<td>While GC idle</td>
<td>35709</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>175</td>
</tr>
<tr>
<td>While GC idle</td>
<td>6138</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>175</td>
</tr>
<tr>
<td>While GC idle</td>
<td>6138</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>21</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>21</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec:</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>8</td>
</tr>
<tr>
<td>While GC idle</td>
<td>748</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>8</td>
</tr>
<tr>
<td>While GC idle</td>
<td>748</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
</tbody>
</table>

106
<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>allocDSsubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>allocTsubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>allocDstack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>40609</td>
<td>68.0996</td>
<td>36.258</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>5754450</td>
<td>58.3505</td>
<td>30.5398</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>32165</td>
<td>26.3949</td>
<td>5.99283</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>5135056</td>
<td>23.9189</td>
<td>6.45983</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ScmBalance/Cycles</td>
<td>76132</td>
<td>0.335761</td>
<td>0.436942</td>
<td>0.000144186</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Heap Utilization</td>
<td>2</td>
<td>0.0123062</td>
<td>0.0149598</td>
<td>0.001728606</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 36664

<table>
<thead>
<tr>
<th></th>
<th>10843898</th>
</tr>
</thead>
<tbody>
<tr>
<td>total cycles required for GC:</td>
<td></td>
</tr>
</tbody>
</table>

bus utilization: 32.32%
utilization due to cache invalidation requests: 0.002%
icache hit rate ( 689131327 hits / 708252855 fetches): 97.300%
of 1.91215e+07 misses, average icache miss cost: 7.500
of 1.91215e+07 Traditional misses, average icache miss cost: 7.500
dcache hit rate ( 65649965 hits / 122071326 fetches): 53.780%
of 5.64216e+07 misses, average dcache miss cost: 3.050
of 368883 GC misses, average dcache miss cost: 23.771

107
of 52063 Traditional misses, average dcache miss cost: 12.502

Simulation completed on Mon Jul 20 01:26:43 1992
CPU usage: 612.8 system, 116783.9 user

3.40 Scaling experiment: 0x40000

dlxsimgc release:
   $Id: iden.t.h,v 1.18 1992/07/16 16:13:19 kelvin Exp $

Command line: /home/rtgc/dlxsim++/dlxsimgc.40000 -a
   /home/rtgc/dlxsim++/lispPUS __start

This simulation begun on Tue Jul 21 11:54:39 1992

Statistics:
   Program image ends at 0x29420

| total machine instructions executed: | 1018666800 |
| cycles stalled for instruction fetch: | 250179291 |
| cycles stalled for memory operations: | 1178932123 |
| cycles stalled following loads: | 641255198 |
| cycles stalled for floating point results: | 0 |
| cycles stalled for floating point processors: | 0 |
| cycles stalled for branch-delay instruction fetches: | 81329 |
| cycles stalled for trap interfacing: | 29995 |

| total machine cycles executed: | 3089144736 |
| total number of traps executed: | 1027 |

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2612</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2612</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>46144</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5402</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>46144</td>
</tr>
<tr>
<td>While GC idle</td>
<td>5402</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>initBlock</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>78</td>
</tr>
<tr>
<td>While GC idle</td>
<td>779</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>78</td>
</tr>
<tr>
<td>While GC idle</td>
<td>779</td>
</tr>
<tr>
<td>tendDesc</td>
<td>39161</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>39161</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
</tbody>
</table>
**While GC idle**: 0 0 0 0 0 0

<table>
<thead>
<tr>
<th>stackPush</th>
<th>Latencies:</th>
<th>Costs:</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>7702615</td>
<td>63.0673</td>
</tr>
<tr>
<td>While GC idle</td>
<td>850423</td>
<td>61.8145</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>stackPop</th>
<th>Latencies:</th>
<th>Costs:</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>7702615</td>
<td>59.1253</td>
</tr>
<tr>
<td>While GC idle</td>
<td>850423</td>
<td>58.2775</td>
</tr>
</tbody>
</table>

**ScanBalance/Cycles**: 17945923 0.258202 3.450210.000256312 1283  
**Heap Utilization**: 2611 0.002768850.0008981460.000793457 0.0109558

* A value of -1 signifies that the mean is not available due to unclaimed results.

- number of allocations unimpeded by GC: 52360  
- total cycles required for GC: 2797451267  
- bus utilization: 49.67%  
- utilization due to cache invalidation requests: 2.915%  
- icache hit rate (1209934524 hits / 120999542 fetches): 99.995%  
  - of 61927 misses, average icache miss cost: 8.723  
  - of 61927 Traditional misses, average icache miss cost: 8.723  
- dcache hit rate ( 50017795 hits / 201780803 fetches): 24.788%  
  - of 1.51763e+08 misses, average dcache miss cost: 9.763  
  - of 5.51522e+07 GC misses, average dcache miss cost: 22.551  
  - of 17105 Traditional misses, average dcache miss cost: 13.251

Simulation completed on Thu Jul 23 21:44:54 1992  
CPU usage: 1275.10 system, 153638.8 user

### 3.41 Scaling experiment: 0x80000

dlxsimc release:

Command line: /home/rtgc/dlxsimc++/dlxsimc.80000 -a  
/home/rtgc/dlxsimc++/lispPUSS._start

This simulation begun on Tue Jul 21 11:43:07 1992

Statistics:
- Program image ends at 0x29420
  - total machine instructions executed: 1000550455  
  - cycles stalled for instruction fetch: 254344390  
  - cycles stalled for memory operations: 1038999112  
  - cycles stalled following loads: 585146709  
  - cycles stalled for floating point results: 0  
  - cycles stalled for floating point processors: 0  
  - cycles stalled for branch-delay instruction fetches: 13299  
  - cycles stalled for trap interfacing: 27894

- total machine cycles executed: 2879081859
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>8</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>8</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>181</td>
</tr>
<tr>
<td>While GC idle</td>
<td>48761</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>181</td>
</tr>
<tr>
<td>While GC idle</td>
<td>48761</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>5</td>
</tr>
<tr>
<td>While GC idle</td>
<td>852</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>5</td>
</tr>
<tr>
<td>While GC idle</td>
<td>852</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>116</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>116</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>5</td>
</tr>
<tr>
<td>While GC idle</td>
<td>808</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>5</td>
</tr>
<tr>
<td>While GC idle</td>
<td>808</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td>25748</td>
</tr>
<tr>
<td>While GC idle</td>
<td>8527290</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>25748</td>
</tr>
<tr>
<td>While GC idle</td>
<td>8527290</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td>25181</td>
</tr>
<tr>
<td>While GC idle</td>
<td>8322482</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>25181</td>
</tr>
<tr>
<td>While GC idle</td>
<td>8322482</td>
</tr>
<tr>
<td>ScanBalance/Cycles</td>
<td>65428</td>
</tr>
<tr>
<td>Heap Utilization</td>
<td>8</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 49756
total cycles required for GC: 9316934

bus utilization: 47.603%
utilization due to cache invalidation requests: 3.023%
icache hit rate (1187380813 hits / 1187397232 fetches): 99.999%
of 16418 misses, average icache miss cost: 7.615
of 16418 Traditional misses, average icache miss cost: 7.615
dcache hit rate (51703497 hits / 197262425 fetches): 26.211%
of 1.45559e+08 misses, average dcache miss cost: 8.892
of 5.3462e+07 GC misses, average dcache miss cost: 20.015
of 13715 Traditional misses, average dcache miss cost: 11.846

Simulation completed on Thu Jul 23 18:48:21 1992
CPU usage: 1742.4 system, 178617.3 user

3.42 Scaling experiment: 0x100000

dlxsimgc release:

Command line: /home/rtgc/dlxsim++/dlxsimgc.100000 -a
/home/rtgc/dlxsim++/lispPUS ___start

This simulation begun on Tue Jul 21 11:46:41 1992

Statistics:
Program image ends at 0x29420

<table>
<thead>
<tr>
<th>operations executed</th>
<th>989995692</th>
</tr>
</thead>
<tbody>
<tr>
<td>cycles stalled for instruction fetch</td>
<td>245922450</td>
</tr>
<tr>
<td>cycles stalled for memory operations</td>
<td>933575766</td>
</tr>
<tr>
<td>cycles stalled for floating point results</td>
<td>550011094</td>
</tr>
<tr>
<td>cycles stalled for floating point processors</td>
<td>0</td>
</tr>
<tr>
<td>cycles stalled for branch-delay instruction fetches</td>
<td>12566</td>
</tr>
<tr>
<td>cycles stalled for trap interfacing</td>
<td>26822</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>total machine cycles executed</th>
<th>2719844330</th>
</tr>
</thead>
<tbody>
<tr>
<td>total number of traps executed</td>
<td>1027</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation invocations</th>
<th>number</th>
<th>mean*</th>
<th>std dev</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td>copyBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
<td>-1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
<td>-1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>59</td>
<td>42.6441</td>
<td>7.26532</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>48877</td>
<td>42.6064</td>
<td>7.74994</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>59</td>
<td>35</td>
<td>6.52581</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>48877</td>
<td>33.1828</td>
<td>5.37197</td>
<td></td>
<td></td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

113
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

initBlock
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

tendDesc
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

allocRec
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

allocDSlice
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

allocTSlice
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

allocDSSubSlice
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

allocTSubSlice
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

allocDStack
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

allocTStack
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
While GC idle | 0 | 0 | 0 | 0 |
Costs: | | | | |
During GC | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0
stackPush Latencies: |
During GC | 8033 | 62.367 | 19.8286 |
While GC idle | 8545005 | 60.3095 | 18.9861 |
Costs: | | | | |
During GC | 8033 | 58.5994 | 19.1879 | 26 | 136
While GC idle | 8545005 | 55.8887 | 19.0893 | 21 | 142
stackPop Latencies: |
During GC | 7813 | 26.6397 | 3.54895 |
While GC idle | 8339850 | 21.9724 | 4.65098 |
Costs: | | | | |
During GC | 7813 | 21.4873 | 3.13026 | 15 | 28
While GC idle | 8339850 | 18.408 | 2.82667 | 15 | 28
ScanBalance/Cycles | 21586 | 0.517977 | 0.783786 | 0.000511509 | 68.8289
Heap Utilization | 2 | 0.001129158 | 0.63167e-05 | 0.001068121 | 0.00119019
* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 49750
total cycles required for GC: 2870815
bus utilization: 45.212%
utilization due to cache invalidation requests: 3.198%
icache hit rate (1174190003 hits / 1174206018 fetches): 99.999%
of 16014 misses, average icache miss cost: 7.507
of 16014 Traditional misses, average icache miss cost: 7.507
dcache hit rate (51723335 hits / 194623760 fetches): 26.576%
of 1.429e+08 misses, average dcache miss cost: 8.256
of 5.34443e+07 GC misses, average dcache miss cost: 18.028
of 11594 Traditional misses, average dcache miss cost: 11.518

Simulation completed on Thu Jul 23 00:22:50 1992
CPU usage: 1037.4 system, 126369.8 user

3.43 Scaling experiment: 0x200000

dlxsimgc release:

Command line: /home/rtgc/dlxsimg+/dlxsimgc.200000 -a
/home/rtgc/dlxsimg+/lispUS __start

This simulation begun on Wed Jul 22 16:11:08 1992

Statistics:
Program image ends at 0x29420
total machine instructions executed: 970017623
cycles stalled for instruction fetch: 229628827
cycles stalled for memory operations: 73499367
cycles stalled following loads: 484285599
cycles stalled for floating point results: 0
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 12152
cycles stalled for trap interfacing: 23794
total machine cycles executed: 2418967362
total number of traps executed: 1027

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>1</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>1</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>37</td>
</tr>
<tr>
<td>While GC idle</td>
<td>48898</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>37</td>
</tr>
<tr>
<td>While GC idle</td>
<td>48898</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>857</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>857</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>15</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>15</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>813</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>813</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>5421</td>
</tr>
<tr>
<td>While GC idle</td>
<td>8547617</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>5421</td>
</tr>
<tr>
<td>While GC idle</td>
<td>8547617</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>5262</td>
</tr>
<tr>
<td>While GC idle</td>
<td>8342401</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>5262</td>
</tr>
<tr>
<td>While GC idle</td>
<td>8342401</td>
</tr>
<tr>
<td>ScmBalance/Cycles</td>
<td>16036</td>
</tr>
<tr>
<td>Heap Utilization</td>
<td>10</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 49749

total cycles required for GC: 1911786

bus utilization: 39.826%

utilization due to cache invalidation requests: 3.596%
icache hit rate (1149221614 hits / 1149237553 fetches): 99.999%
of 15938 misses, average icache miss cost: 7.304
of 15938 Traditional misses, average icache miss cost: 7.304
dcache hit rate (51723453 hits / 189629248 fetches): 27.27%
of 1.37906e+08 misses, average dcache miss cost: 6.991
of 5.34447e+07 GC misses, average dcache miss cost: 14.270
of 11050 Traditional misses, average dcache miss cost: 11.218

Simulation completed on Fri Jul 24 02:59:05 1992
CPU usage: 998.6 system, 119688.6 user

3.44 Scaling experiment: 0x400000

dlxsimgc release:

Command line: /home/rtgc/dlxsimg++/dlxsimgc.400000 -a
/home/rtgc/dlxsimg++/lispPUS__start

This simulation begun on Fri Jul 24 09:59:14 1992

Statistics:
   Program image ends at 0x29420

   total machine instructions executed: 963561584
   cycles stalled for instruction fetch: 224381609
   cycles stalled for memory operations: 670669721
   cycles stalled following loads: 462886334
   cycles stalled for floating point results: 0
   cycles stalled for floating point processors: 0
   cycles stalled for branch-delay instruction fetches: 11821
   cycles stalled for trap interfacing: 23507
   -----------------------------
   total machine cycles executed: 2321534576
   total number of traps executed: 1027

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operations</th>
<th>number</th>
<th>mean*</th>
<th>std dev</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
|          | Costs: | During GC| 0 | 0 | 0 | 0 | 0
|          | While GC idle| 0 | 0 | 0 | 0 | |
| tendingDone | Latencies: | During GC| 0 | 0 | 0 | |
|          | While GC idle| 0 | 0 | 0 | |
|          | Costs: | During GC| 0 | 0 | 0 | 0 | 0
<p>|          | While GC idle| 0 | 0 | 0 | 0 | |
| allocInitRec | Latencies: | During GC| 0 | 0 | 0 | |
|          | While GC idle| 48934 | 40.9444 | 8.43102 | |</p>
<table>
<thead>
<tr>
<th>Method</th>
<th>Duration (ms)</th>
<th>CPU (ms)</th>
<th>Latencies</th>
<th>Costs</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>48934</td>
<td>30.1288</td>
<td>4.75575</td>
<td>14</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>857</td>
<td>20.8203</td>
<td>26.9054</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>857</td>
<td>17.6756</td>
<td>26.4512</td>
<td>8</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>813</td>
<td>19.0295</td>
<td>1.26301</td>
<td></td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>813</td>
<td>11</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>13</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
<td>12</td>
<td>0</td>
<td>12</td>
</tr>
</tbody>
</table>

Number 119
<table>
<thead>
<tr>
<th>allocTStack</th>
<th>Latencies:</th>
<th>Costs:</th>
<th>stackPush</th>
<th>Latencies:</th>
<th>Costs:</th>
<th>stackPop</th>
<th>Latencies:</th>
<th>Costs:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>8553038</td>
<td>58.8283</td>
<td>18.5497</td>
<td>8347663</td>
<td>16.3834</td>
<td>1.81758</td>
<td>8347663</td>
<td>15.2543</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x8f82f8)

number of allocations unimpeded by GC: 49748

total cycles required for GC: 0

bus utilization: 37.783%

total cycles due to cache invalidation requests: 3.746%
icache hit rate (141153072 hits / 1441169851 fetches): 99.999%
of 15778 misses, average icache miss cost: 7.204

do of 15778 Traditional misses, average icache miss cost: 7.204
dcache hit rate (51729354 hits / 188015248 fetches): 27.513%
of 1.36286e+08 misses, average dcache miss cost: 6.561

do 5.34402e+07 GC misses, average dcache miss cost: 13.054

do 9688 Traditional misses, average dcache miss cost: 10.656

Simulation completed on Sat Jul 25 19:18:01 1992

CPU usage: 10.14.10 system, 113865.1 user

4 Raw Test Results for Chapter 6 Experiments

4.1 Experimental results for SS compiler

4.1.1 sfft/small

dlxsimgc release:

$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp$

Command line: /usr/schmidt/gmugc/sim/slowcpu.morestats/dlxsimgc.100000
-a sfft _start ..data/small

This simulation begun on Fri Jul 24 16:49:14 1992

Statistics:

Program image ends at 0x35740
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Operations</td>
</tr>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendDone</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendDesc</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>1</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>105790</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>105790</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>68889</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>68889</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.
warning: no garbage collection activity (New = 0x86dd38)

number of allocations unimpeded by GC: 3
total cycles required for GC: 0

bus utilization: 38.275%
utilization due to cache invalidation requests: 1.950%
icache hit rate ( 76561613 hits / 76568318 fetches): 99.991%
dcache hit rate ( 20367275 hits / 23607099 fetches): 86.276%

Simulation completed on Sat Jul 25 05:36:26 1992
CPU usage: 30.9 system, 11102.2 user

4.1.2 sfft/medium
dlxsimc release:

Command line: dlxsimc.100000 -a sfftPIF --start medium

This simulation begun on Sat Aug 1 10:40:33 1992

Statistics:
Program image ends at 0x35740

total machine instructions executed: 281423918
cycles stalled for instruction fetch: 19373501
cycles stalled for memory operations: 136221812
cycles stalled following loads: 123012277
cycles stalled for floating point results: 5773023
cycles stalled for floating point processors: 65535
cycles stalled for branch-delay instruction fetches: 4218
cycles stalled for trap interfacing: 7936

---------
total machine cycles executed: 565982220

---------
total number of traps executed: 423

--------------------------------------------------------------------------
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>number</td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
</tr>
<tr>
<td>Operations</td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>724</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>724</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>270586</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>270586</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>2</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocDSStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
</tbody>
</table>
While GC idle| 1 | 12 | 0 | 12 | 12
allocTStack
Latencies:
| | | | | |
| During GC| 0 | 0 | 0 | |
| While GC idle| 0 | 0 | 0 | |
Costs:
| | | | | |
| During GC| 0 | 0 | 0 | 0 | 0
| While GC idle| 0 | 0 | 0 | 0 | 0
stackPush
Latencies:
| | | | | |
| During GC| 0 | 0 | 0 | |
| While GC idle| 379737 | 101.661 | 22.5667 | |
Costs:
| | | | | |
| During GC| 0 | 0 | 0 | 0 | 0
| While GC idle| 379737 | 96.4938 | 22.807 | 30 | 127
stackPop
Latencies:
| | | | | |
| During GC| 0 | 0 | 0 | |
| While GC idle| 270096 | 16.0235 | 0.458885 | |
Costs:
| | | | | |
| During GC| 0 | 0 | 0 | 0 | 0
| While GC idle| 270096 | 15.0206 | 0.404904 | 15 | 23

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x86dd38)

number of allocations unimpeded by GC: 3
total cycles required for GC: 0

bus utilization: 38.438%
utilization due to cache invalidation requests: 1.753%
icache hit rate (300830600 hits / 300838020 fetches): 99.996%
of 7419 misses, average icache miss cost: 6.931
do of 7419 Traditional misses, average icache miss cost: 6.931
dcache hit rate (81272843 hits / 93433826 fetches): 86.984%
of 1.216e+07 misses, average dcache miss cost: 6.594
do of 5.65813e+06 GC misses, average dcache miss cost: 11.121
do of 271341 Traditional misses, average dcache miss cost: 11.574

Simulation completed on Sun Aug 2 02:49:11 1992
CPU usage: 14.7 system, 28461.7 user

4.1.3 lisp/db
dlxsimgc release:

Command line: slow/dlxsimgc.1000000 -a lispPIT __start
This simulation begun on Thu Jul 2 11:13:31 1992
Statistics:
Program image ends at 0x2e870
total machine instructions executed: 661419483
cycles stalled for instruction fetch: 128751590
cycles stalled for memory operations: 505726061
cycles stalled following loads: 392923863
cycles stalled for floating point results: 0

125
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 91950
cycles stalled for trap interfacing: 19787

total machine cycles executed: 1685932734

total number of traps executed: 952

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Operations</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>copyPush</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>allocInitRec</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>allocRec</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>allocDSSlice</td>
</tr>
</tbody>
</table>

126
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>3007</td>
<td>57.86</td>
<td>12.7535</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>4694397</td>
<td>54.3298</td>
<td>11.38</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>3007</td>
<td>53.3838</td>
<td>11.3198</td>
<td>38</td>
<td>118</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>4694397</td>
<td>48.9594</td>
<td>11.6304</td>
<td>21</td>
<td>137</td>
<td></td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>2995</td>
<td>26.3656</td>
<td>3.39085</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>4676989</td>
<td>19.2841</td>
<td>4.53461</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>During GC</td>
<td>2995</td>
<td>21.3491</td>
<td>2.96485</td>
<td>15</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>4676989</td>
<td>16.8848</td>
<td>2.72304</td>
<td>15</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>ScmBalance/Cycles</td>
<td>10769</td>
<td>0.556385</td>
<td>4.22119</td>
<td>0.00156207</td>
<td>345.917</td>
<td></td>
</tr>
<tr>
<td>Heap Utilization</td>
<td>1</td>
<td>0.000991821</td>
<td>0</td>
<td>0.000991821</td>
<td>0.000991821</td>
<td></td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 28936
total cycles required for GC: 1294823
bus utilization: 38.53%
utilization due to cache invalidation requests: 2.973%
icache hit rate ( 783251444 hits / 783353723 fetches): 99.987%
dcache hit rate ( 35991770 hits / 12486293 fetches): 28.817%

Simulation completed on Sun Jul 5 13:19:20 1992
CPU usage: 862.3 system, 176357.4 user

4.1.4 lisp/prune
dlxisimc release:
$Id: ident.hh,v 1.14 1992/06/20 17:28:45 kelvin Exp$
Command line: dlxisimc.80000 -a lispPIF _start
This simulation begun on Sun Jun 21 12:35:52 1992
Statistics:
Program text ranges from 0x100 to 0x168e0

<table>
<thead>
<tr>
<th></th>
<th>number</th>
<th>mean*</th>
<th>std dev</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
<td>-1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2</td>
<td>-1</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>25</td>
<td>36</td>
<td>5.95819</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>18472</td>
<td>35.4618</td>
<td>5.88732</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>25</td>
<td>30.72</td>
<td>3.97408</td>
<td>19</td>
</tr>
<tr>
<td>While GC idle</td>
<td>18472</td>
<td>28.432</td>
<td>4.49577</td>
<td>5</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Function</td>
<td>Latencies (GC, idle)</td>
<td>Costs (GC, idle)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>----------------------</td>
<td>------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Costs:

- During GC: 0
- While GC idle: 0

**initBlock**

<table>
<thead>
<tr>
<th>Latencies (GC, idle)</th>
<th>Costs (GC, idle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>3276 22.5833 5.33072</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2551901 21.2204 4.88202</td>
</tr>
</tbody>
</table>

Costs:

- During GC: 3276
- While GC idle: 2551901

**tendDesc**

<table>
<thead>
<tr>
<th>Latencies (GC, idle)</th>
<th>Costs (GC, idle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>30 32.1667 13.5369</td>
</tr>
<tr>
<td>While GC idle</td>
<td>301 19.0199 1.50983</td>
</tr>
</tbody>
</table>

Costs:

- During GC: 30
- While GC idle: 301

**allocRec**

<table>
<thead>
<tr>
<th>Latencies (GC, idle)</th>
<th>Costs (GC, idle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0 0 0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0 0 0</td>
</tr>
</tbody>
</table>

Costs:

- During GC: 0
- While GC idle: 0

**allocDSlice**

<table>
<thead>
<tr>
<th>Latencies (GC, idle)</th>
<th>Costs (GC, idle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0 0 0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>301 11.6977 1.7354</td>
</tr>
</tbody>
</table>

Costs:

- During GC: 0
- While GC idle: 301

**allocTSlice**

<table>
<thead>
<tr>
<th>Latencies (GC, idle)</th>
<th>Costs (GC, idle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0 0 0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0 0 0</td>
</tr>
</tbody>
</table>

Costs:

- During GC: 0
- While GC idle: 0

**allocDSubSlice**

<table>
<thead>
<tr>
<th>Latencies (GC, idle)</th>
<th>Costs (GC, idle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0 0 0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0 0 0</td>
</tr>
</tbody>
</table>

Costs:

- During GC: 0
- While GC idle: 0

**allocTSubSlice**

<table>
<thead>
<tr>
<th>Latencies (GC, idle)</th>
<th>Costs (GC, idle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0 0 0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0 0 0</td>
</tr>
</tbody>
</table>

Costs:

- During GC: 0
- While GC idle: 0

**allocDStack**

<table>
<thead>
<tr>
<th>Latencies (GC, idle)</th>
<th>Costs (GC, idle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>1 13 0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>17 13 0</td>
</tr>
</tbody>
</table>

Costs:

- During GC: 1
- While GC idle: 17

**allocTStack**

<table>
<thead>
<tr>
<th>Latencies (GC, idle)</th>
<th>Costs (GC, idle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>During GC</td>
<td>0 0 0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0 0 0</td>
</tr>
</tbody>
</table>

Costs:
4.2 Experimental results for SU compiler

4.2.1 sftt/small

dlxsimgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp$

Command line: /usr/schmidt/gnu/gc/sim/slowcpu.morestats/dlxsimgc.100000
-a sftt__start .../data/small

This simulation begun on Fri Jul 24 16:56:54 1992

Statistics:
  Program image ends at 0x32860

  total machine instructions executed: 70747535
  cycles stalled for instruction fetch: 4985475
  cycles stalled for memory operations: 34309027
  cycles stalled following loads: 31259452
  cycles stalled for floating point results: 1456969
  cycles stalled for floating point processors: 16383
  cycles stalled for branch-delay instruction fetches: 4263
  cycles stalled for trap interfacing: 4610

  total machine cycles executed: 142783714
  total number of traps executed: 199
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>345</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>345</td>
</tr>
<tr>
<td>tendingDone Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>copyPush Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>71</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>71</td>
</tr>
<tr>
<td>tendDesc Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2</td>
</tr>
<tr>
<td>allocDSlice Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
</tbody>
</table>
During GC | 0 | 0 | 0 |
While GC idle | 0 | 0 | 0 |
Costs:
  During GC | 0 | 0 | 0 | 0 | 0
  While GC idle | 0 | 0 | 0 | 0 | 0
allocDSubSlice |
Latencies:
  During GC | 0 | 0 | 0 |
  While GC idle | 0 | 0 | 0 |
Costs:
  During GC | 0 | 0 | 0 | 0 | 0
  While GC idle | 0 | 0 | 0 | 0 | 0
allocDStack |
Latencies:
  During GC | 0 | 0 | 0 |
  While GC idle | 0 | 0 | 0 |
Costs:
  During GC | 0 | 0 | 0 | 0 | 0
  While GC idle | 0 | 0 | 0 | 0 | 0
stackPush |
Latencies:
  During GC | 0 | 0 | 0 |
  While GC idle | 122365 | 105.879 | 32.3833 |
Costs:
  During GC | 0 | 0 | 0 | 0 | 0
  While GC idle | 122365 | 100.462 | 33.0613 | 29 | 127
stackPop |
Latencies:
  During GC | 0 | 0 | 0 |
  While GC idle | 68889 | 16.0423 | 0.615771 |
Costs:
  During GC | 0 | 0 | 0 | 0 | 0
  While GC idle | 68889 | 15.0367 | 0.538595 | 15 | 23

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x86dd38)

number of allocations unimpeded by GC: 3
total cycles required for GC: 0

bus utilization: 38.634%
utilization due to cache invalidation requests: 2.101%
icache hit rate ( 75726358 hits / 75733489 fetches): 99.991%
dcache hit rate ( 20279754 hits / 23561939 fetches): 86.070%

Simulation completed on Sat Jul 25 05:37:16 1992
CPU usage: 31.6 system, 10886.9 user
### 4.2.2 ssft/medium

**dlxsimc release:**
```
```

**Command line:** dlxsimc.100000 -a ssftPUS --start medium

This simulation begun on Sat Aug 1 10:40:48 1992

**Statistics:**
Program image ends at 0x32860

- Total machine instructions executed: 278603605
- Cycles stalled for instruction fetch: 18959338
- Cycles stalled for memory operations: 136118865
- Cycles stalled following loads: 122406353
- Cycles stalled for floating point results: 5773033
- Cycles stalled for floating point processors: 65535
- Cycles stalled for branch-delay instruction fetches: 6242
- Cycles stalled for trap interfacing: 10014

Total machine cycles executed: 561942985

Total number of traps executed: 423

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>724</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>724</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>----------</td>
<td>---------------</td>
</tr>
<tr>
<td></td>
<td>71</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tmdDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
</tbody>
</table>
During GC | 0 | 0 | 0 | 0 | 0
While GC idle | 445687 | 98.703 | 33.2908 | 29 | 127

Latencies:

| During GC | 0 | 0 | 0 | 0 | 0
While GC idle | 270096 | 16.0235 | 0.458885 | 15 | 23

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x86dd38)

number of allocations unimpeded by GC: 3
total cycles required for GC: 0

bus utilization: 38.794%
utilization due to cache invalidation requests: 1.904%
icache hit rate ( 29763945 hits / 29765316 fetches): 99.99%
of 9470 misses, average icache miss cost: : 7.293
of 9470 Traditional misses, average icache miss cost: : 7.293
dcache hit rate ( 80932612 hits / 93272363 fetches): 86.770%
of 1.23402e+07 misses, average dcache miss cost: : 6.448
of 5.52146e+06 GC misses, average dcache miss cost: : 11.302
of 271871 Traditional misses, average dcache miss cost: : 11.552

Simulation completed on Sun Aug 2 02:40:40 1992
CPU usage: 13.7 system, 27953.4 user

4.2.3 lisp/db
dlxsimc release:

Command line: slow/dlxsimc.100000 -a lispPUS __start
This simulation begun on Thu Jul 2 11:09:27 1992

Statistics:
Program image ends at 0x29410
total machine instructions executed: 548617448
cycles stalled for instruction fetch: 132351476
cycles stalled for memory operations: 454668675
cycles stalled following loads: 285880268
cycles stalled for floating point results: 0
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 10944
cycles stalled for trap interfacing: 23722
----------
total machine cycles executed: 1421740733
total number of traps executed: 948

Arbiter | Operation Invocations
---------|-------------------------
Operations | number | mean* | std dev | range

135
<table>
<thead>
<tr>
<th></th>
<th>copyBlock</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>31</td>
<td>42.6129</td>
<td>7.21423</td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>28128</td>
<td>41.6173</td>
<td>8.05959</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>31</td>
<td>34.7097</td>
<td>6.16546</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>28128</td>
<td>31.6813</td>
<td>5.34525</td>
<td>5</td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>776</td>
<td>19.0309</td>
<td>1.29279</td>
<td></td>
</tr>
<tr>
<td>allocDSlice</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>776</td>
<td>11.0644</td>
<td>0.56429</td>
<td>11</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

136
Latencies:

<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>allocTSubSlice</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Costs:

<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>allocTSubSlice</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Latencies:

<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>allocDStack</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Costs:

<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>allocDStack</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Latencies:

<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>stackPush</td>
<td>3920</td>
<td>62.3421, 19.8937</td>
</tr>
</tbody>
</table>

Costs:

<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>stackPush</td>
<td>4797853</td>
<td>59.7272, 18.9846</td>
</tr>
</tbody>
</table>

Latencies:

<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>stackPop</td>
<td>3810</td>
<td>26.569, 3.5051</td>
</tr>
</tbody>
</table>

Costs:

<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>stackPop</td>
<td>4676166</td>
<td>19.3531, 4.56228</td>
</tr>
</tbody>
</table>

ScmBalance/Cycles:

|                        | 10761 | 0.520152 | 0.806466 | 0.00954665 | 68.8289 |

Heap Utilization:

|                        | 0.00119019 | 0.00119019 | 0.00119019 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 28936

total cycles required for GC: 1400461

bus utilization: 42.014%

utilization due to cache invalidation requests: 3.447%

icache hit rate (650482221 hits / 650496300 fetches): 99.996%

dcache hit rate (29022250 hits / 107531988 fetches): 26.988%

Simulation completed on Sat Jul 4 16:47:29 1992

CPU usage: 481.8 system, 115739.6 user

4.2.4 lisp/prune

dlxsimgc release:

Command line: slow/dlxsimgc.80000 -a lispPUS _start

This simulation begun on Thu Jul  2 11:09:57 1992

Statistics:
Program image ends at 0x29410

total machine instructions executed: 303278183
cycles stalled for instruction fetch: 73706804
cycles stalled for memory operations: 284505090
cycles stalled following loads: 170801574
cycles stalled for floating point results: 0
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 9241
cycles stalled for trap interfacing: 1927

----
total machine cycles executed: 832302819

----
total number of traps executed: 82

-----------------------------------------------
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Operations</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------</td>
</tr>
<tr>
<td></td>
<td>copyBlock</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
</tbody>
</table>

138
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>30</td>
<td>29.9</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>18.5</td>
</tr>
<tr>
<td>allocRec</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>301</td>
<td>19.1395</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>301</td>
<td>11.6977</td>
</tr>
<tr>
<td>allocDSlice</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>allocTStack</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td>4661</td>
<td>62.1352</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4661</td>
<td>58.44</td>
</tr>
<tr>
<td>stackPop</td>
<td>2620087</td>
<td>55.8832</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4530</td>
<td>3.17234</td>
</tr>
</tbody>
</table>
4.3 Experimental results for HS compiler

4.3.1 sfft/small

dlxsimgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line: /usr/schmidt/gnu/sim/slowcpu.morestats/dlxsimgc.100000 -a sfft __start .../data/small

This simulation begun on Fri Jul 24 16:58:45 1992

Statistics:
Program image ends at 0x34ec0

total machine instructions executed: 70797903
cycles stalled for instruction fetch: 4649800
cycles stalled for memory operations: 67700112
cycles stalled following loads: 47440796
cycles stalled for floating point results: 1456933
cycles stalled for floating point processors: 16383
cycles stalled for branch-delay instruction fetches: 4796
cycles stalled for trap interfacing: 6190

total machine cycles executed: 192072913

total number of traps executed: 199
<table>
<thead>
<tr>
<th></th>
<th>296</th>
<th>68.5878</th>
<th>26.4416</th>
<th>29</th>
<th>110</th>
</tr>
</thead>
<tbody>
<tr>
<td>tendingDone</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>24</td>
<td>-1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>24</td>
<td>-1</td>
<td>0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2823</td>
<td>67.4013</td>
<td>414.397</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>49184</td>
<td>48.9976</td>
<td>2.19327</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>2823</td>
<td>60.7287</td>
<td>414.263</td>
<td>31</td>
<td>12539</td>
</tr>
<tr>
<td>While GC idle</td>
<td>49184</td>
<td>42.7562</td>
<td>2.07201</td>
<td>5</td>
<td>54</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>8460</td>
<td>80.1431</td>
<td>51.0505</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>114489</td>
<td>65.2026</td>
<td>48.5198</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>8460</td>
<td>75.7652</td>
<td>50.7574</td>
<td>11</td>
<td>127</td>
</tr>
<tr>
<td>While GC idle</td>
<td>114489</td>
<td>60.8725</td>
<td>48.5127</td>
<td>11</td>
<td>172</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>252</td>
<td>21.873</td>
<td>13.6449</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>252</td>
<td>17.6071</td>
<td>12.1186</td>
<td>3</td>
<td>32</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>911</td>
<td>3083.87</td>
<td>10193.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>15998</td>
<td>18.9955</td>
<td>0.268316</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>911</td>
<td>3080.48</td>
<td>10192</td>
<td>11</td>
<td>110186</td>
</tr>
<tr>
<td>While GC idle</td>
<td>15998</td>
<td>15.9136</td>
<td>0.691686</td>
<td>5</td>
<td>16</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Latencies:      |      |      |      |      |
  During GC      0    0    0    0    0
  While GC idle  0    0    0    0    0
Costs:         |      |      |      |      |
  During GC      0    0    0    0
  While GC idle  0    0    0    0
allocDSStack    |      |      |      |      |
  Latencies:     |      |      |      |      |
  During GC      0    0    0    0
  While GC idle  0    0    0    0
Costs:         |      |      |      |      |
  During GC      0    0    0    0
  While GC idle  0    0    0    0
allocTStack     |      |      |      |      |
  Latencies:     |      |      |      |      |
  During GC      0    0    0    0
  While GC idle  0    0    0    0
stackPush       |      |      |      |      |
  Latencies:     |      |      |      |      |
  During GC      0    0    0    0
  While GC idle  0    0    0    0
Costs:         |      |      |      |      |
  During GC      0    0    0    0
  While GC idle  0    0    0    0
stackPop        |      |      |      |      |
  Latencies:     |      |      |      |      |
  During GC      0    0    0    0
  While GC idle  0    0    0    0
Costs:         |      |      |      |      |
  During GC      0    0    0    0
  While GC idle  0    0    0    0
ScanBalance/Cycles  126171  1.01184  1.849  0.0036036  290.611
Heap Utilization  24  0.0649236  0.035146  0.0438385  0.129013

* A value of -1 signifies that the mean is not available due
to unclaimed results.

number of allocations unimpeded by GC: 68853
total cycles required for GC: 13137459

bus utilization: 54.26%
utilization due to cache invalidation requests: 1.006%
icache hit rate (7536798 hits / 7537981 fetches): 99.990%
dcache hit rate (20270599 hits / 23668227 fetches): 85.648%

Simulation completed on Sat Jul 25 06:33:14 1992
CPU usage: 38.0 system, 12466.8 user

4.3.2 sfft/medium
dlxsimgc release:
Command line: /usr/schmidt/gnuigc/sim/editor/dlxsimgc.100000 -a sfft
--start . /data/medium

This simulation begun on Sat Aug 1 12:40:05 1992

142
Statistics:
Program image ends at 0x34ec0

total machine instructions executed: 276205128
cycles stalled for instruction fetch: 17261337
cycles stalled for memory operations: 269824665
cycles stalled following loads: 185222686
cycles stalled for floating point results: 5772883
cycles stalled for floating point processors: 65535
cycles stalled for branch-delay instruction fetches: 6204
cycles stalled for trap interfacing: 13818

-----------------------------
total machine cycles executed: 754372256
total number of traps executed: 423
-----------------------------

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Invocation Latencies:</td>
</tr>
<tr>
<td></td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendingDone</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendDesc</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>-------</td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDSlice Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocTSlice Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDSSubSlice Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocTSSubSlice Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDSStack Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocTStack Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>stackPush Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>stackPop Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ScanBalance/Cycles</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>446801</td>
<td>1.01597</td>
<td>2.77468</td>
<td>0.00035727</td>
</tr>
<tr>
<td>Heap Utilization</td>
<td>85</td>
<td>0.0227718</td>
<td>0.0130038</td>
<td>0.0178452</td>
</tr>
</tbody>
</table>
A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 269995
total cycles required for GC: 46425882

bus utilization: 54.924%
utilization due to cache invalidation requests: 0.853%
icache hit rate (293300737 hits / 293308979 fetches): 99.997%
of 8241 misses, average icache miss cost: 8.093
of 8241 Traditional misses, average icache miss cost: 8.093
dcache hit rate (80889443 hits / 93041834 fetches): 86.936%
of 1.21534e+07 misses, average dcache miss cost: 10.414
of 5.73099e+06 GC misses, average dcache miss cost: 18.426
of 512383 Traditional misses, average dcache miss cost: 14.963

Simulation completed on Sun Aug 2 17:26:38 1992
CPU usage: 124.10 system, 49331.9 user

4.3.3 lisp/db
dlxsimgc release:
Command line: slow/dlxsimgc.100000 -a lispHEAP __start
This simulation begun on Thu Jul 2 11:41:21 1992
Statistics:
  Program image ends at 0x2d930

total machine instructions executed: 624820945
cycles stalled for instruction fetch: 95067677
cycles stalled for memory operations: 64342724
cycles stalled following loads: 467188899
cycles stalled for floating point results: 0
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 181800
cycles stalled for trap interfacing: 31271

-----------------------
total machine cycles executed: 1830717866

total number of traps executed: 948

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>490</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>Latencies:</td>
<td>During GC</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendDesc</td>
<td>Latencies:</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDSSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocTSSubSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>While GC idle</td>
</tr>
</tbody>
</table>
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
allocDStack
Latencies:
| | | | | | |
| During GC | 0 | 0 | 0 | | |
While GC idle | 0 | 0 | 0 | | |
Costs:
| | | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
allocTStack
Latencies:
| | | | | | |
| During GC | 0 | 0 | 0 | | |
While GC idle | 0 | 0 | 0 | | |
Costs:
| | | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
stackPush
Latencies:
| | | | | | |
| During GC | 0 | 0 | 0 | | |
While GC idle | 0 | 0 | 0 | | |
Costs:
| | | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
stackPop
Latencies:
| | | | | | |
| During GC | 0 | 0 | 0 | | |
While GC idle | 0 | 0 | 0 | | |
Costs:
| | | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
ScnBalance/Cycles | 5345450 | 0.555445 | 1.125160.000483676 | 1307.75
Heap Utilization | 490 | 0.131290.0018058 | 0.127754 | 0.134804
-------------------------------
* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 4636729
total cycles required for GC: 443014812
bus utilization: 43.737%
utilization due to cache invalidation requests: 0.767%
icache hit rate ( 714986424 hits / 715166926 fetches): 99.975%
dcache hit rate ( 29914604 hits / 133262472 fetches): 22.448%

Simulation completed on Sat Jul 4 23:35:40 1992
CPU usage: 1240.9 system, 205727.8 user

4.3.4 lisp/prune
dlxsimgc release:
Command line: slow/dlxsimgc.80000 -a lispHEAP __start
This simulation begun on Tue Jun 30 15:31:33 1992
Statistics:
  Program image ends at 0x2d930
total machine instructions executed: 352039396
cycles stalled for instruction fetch: 53421296
cycles stalled for memory operations: 351358066
cycles stalled following loads: 260152123
cycles stalled for floating point results: 0
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 145168
cycles stalled for trap interfacing: 2128

-------------------
total machine cycles executed: 1017118167

-------------------
total number of traps executed: 82

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operations</th>
<th>Invocation</th>
<th>mean*</th>
<th>std dev</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>number</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
<td>tendingDone</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>During GC</td>
<td>565</td>
<td>-1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>During GC</td>
<td>480431</td>
<td>278.146</td>
<td>713.924</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While GC idle</td>
<td>2092998</td>
<td>53.6516</td>
<td>7.29171</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>During GC</td>
<td>480431</td>
<td>267.975</td>
<td>714.583</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While GC idle</td>
<td>2092998</td>
<td>43.1916</td>
<td>6.53988</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While allocRec</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>allocInitRec</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>During GC</td>
<td>477141</td>
<td>21.6582</td>
<td>5.32987</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While GC idle</td>
<td>2080038</td>
<td>21.4018</td>
<td>5.60926</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>During GC</td>
<td>477141</td>
<td>17.6529</td>
<td>3.70462</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While GC idle</td>
<td>2080038</td>
<td>17.4805</td>
<td>4.02404</td>
</tr>
<tr>
<td></td>
<td></td>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>During GC</td>
<td>7261</td>
<td>28.5549</td>
<td>11.9731</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>During GC</td>
<td>7261</td>
<td>24.0846</td>
<td>11.1273</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>During GC</td>
<td>60</td>
<td>4559.72</td>
<td>8795.13</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While GC idle</td>
<td>596</td>
<td>19.0403</td>
<td>1.47531</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>During GC</td>
<td>60</td>
<td>4555.47</td>
<td>8792.44</td>
</tr>
<tr>
<td></td>
<td></td>
<td>While GC idle</td>
<td>596</td>
<td>14.2869</td>
<td>2.39816</td>
</tr>
</tbody>
</table>

148
| allocDSlice | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | |
| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |
| allocTSlice | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | |
| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |
| allocDSubSlice | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | |
| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |
| allocTStack | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | |
| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |
| stackPush | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | |
| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |
| stackPop | | | | |
| Latencies: | | | | |
| During GC | 0 | 0 | 0 | |
| While GC idle | 0 | 0 | 0 | |
| Costs: | | | | |
| During GC | 0 | 0 | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 | 0 | 0 |

ScanBalance/Cycles: 2984941 | 0.568529 | 1.54931 | 0.000480654 | 2566
Heap Utilization: 565 | 0.1372 | 0.00160827 | 0.128189 | 0.140854

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 2521726
4.4 Experimental results for HU compiler

4.4.1 sfft/small

dlxsimgc release:
$Id: ident.hh,v 1.15 1992/06/24 20:11:14 kelvin Exp $

Command line:/usr/schmidt/gnugc/sim/slowcpu.morestats/dlxsimgc.100000
-a sfft _-start ..../data/small

This simulation begun on Sat Jul 25 09:54:29 1992

Statistics:
Program image ends at 0x32020

total machine instructions executed: 69689554
cycles stalled for instruction fetch: 4470286
cycles stalled for memory operations: 67652549
cycles stalled following loads: 46564182
cycles stalled for floating point results: 1456920
cycles stalled for floating point processors: 16383
cycles stalled for branch-delay instruction fetches: 5197
cycles stalled for trap interfacing: 7354

---------------
total machine cycles executed: 189862425

total number of traps executed: 199

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocation Latencies</th>
<th>Operations</th>
<th>number</th>
<th>mean*</th>
<th>std dev</th>
<th>range</th>
</tr>
</thead>
<tbody>
<tr>
<td>copyBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>50</td>
<td>72.46</td>
<td>27.8979</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>295</td>
<td>73.7932</td>
<td>26.7113</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>50</td>
<td>66.92</td>
<td>26.9306</td>
<td>34</td>
<td>102</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>295</td>
<td>68.3966</td>
<td>26.2678</td>
<td>29</td>
<td>102</td>
<td></td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>27</td>
<td>-1</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>27</td>
<td>-1</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>3291</td>
<td>58.9368</td>
<td>134.294</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>48717</td>
<td>49.2539</td>
<td>2.6948</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>3291</td>
<td>52.2549</td>
<td>133.897</td>
<td>37</td>
<td>4600</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>48717</td>
<td>42.9905</td>
<td>2.32585</td>
<td>5</td>
<td>66</td>
<td></td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>5878</td>
<td>115.572</td>
<td>35.6662</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>64562</td>
<td>109.278</td>
<td>40.1717</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>5878</td>
<td>110.242</td>
<td>36.3469</td>
<td>20</td>
<td>127</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>64562</td>
<td>103.846</td>
<td>40.9662</td>
<td>14</td>
<td>172</td>
<td></td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>255</td>
<td>21.9451</td>
<td>13.7565</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>255</td>
<td>17.6902</td>
<td>12.3598</td>
<td>3</td>
<td>32</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>1068</td>
<td>2704.65</td>
<td>9245.96</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>15843</td>
<td>18.9947</td>
<td>0.277912</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>1068</td>
<td>2701.17</td>
<td>9244.81</td>
<td>11</td>
<td>110840</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>15843</td>
<td>15.9116</td>
<td>0.70466</td>
<td>5</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td>allocDSSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
allocTStack  | | | | | |
Latencies:
  Based on GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
Costs:
  Based on GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
stackPush
Latencies:
  During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
Costs:
  During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
stackPop
Latencies:
  During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
Costs:
  During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0
ScannBalance/Cycles | 141924 | 1.00811 | 1.55363 | 0.000352361 | 249.095
Heap Utilization | 27 | 0.0693619 | 0.0289152 | 0.0531768 | 0.128113

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 68826
total cycles required for GC: 14829909
bus utilization: 54.61%
utilization due to cache invalidation requests: 1.066%
Icache hit rate ( 74061620 hits / 74071181 fetches): 99.907%
Dcache hit rate ( 20179629 hits / 23530309 fetches): 85.76%

Simulation completed on Sat Jul 25 16:10:18 1992
CPU usage: 32.7 system, 12260.4 user

4.4.2 sfft/medium
dlxsimgc release:
Command line: /usr/schmidt/gnuC/sim/editor/dlxsimgc.100000 -a sfft
  --start ../data/medium
This simulation begun on Sat Aug 1 12:39:45 1992

Statistics:
  Program image ends at 0x32020
  
  total machine instructions executed: 272012849
cycles stalled for instruction fetch: 16561956
cycles stalled for memory operations: 269732841
cycles stalled following loads: 181967569
cycles stalled for floating point results: 5772849
cycles stalled for floating point processors: 65535
cycles stalled for branch-delay instruction fetches: 7619
cycles stalled for trap interfacing: 16348

----------------
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>---------------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>617</td>
</tr>
<tr>
<td>While GC idlev</td>
<td>107</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>617</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>190633</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>190633</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>223064</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>223064</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>62277</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>62277</td>
</tr>
<tr>
<td>allocDSSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
</tbody>
</table>
While GC idle| 0| 0| 0| 0| 0| 0
allocTSlice
Latencies:
| | | | | | |
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
Costs:
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
allocDSubSlice
Latencies:
| | | | | | |
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
Costs:
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
allocTSubSlice
Latencies:
| | | | | | |
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
Costs:
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
allocTStack
Latencies:
| | | | | | |
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
Costs:
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
stackPush
Latencies:
| | | | | | |
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
Costs:
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
stackPop
Latencies:
| | | | | | |
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
Costs:
| During GC| 0| 0| 0| 0| 0| 0
| While GC idle| 0| 0| 0| 0| 0| 0
ScmBalance/Cycles| 520328| 1.0108| 2.48297| 0.00035057| 1307.75
Heap Utilization| 99| 0.0230247| 0.0129906| 0.0180893| 0.0589523
---------------------------------------------------------------
* A value of -1 signifies that the mean is not available due to unclaimed results.

data of allocations unimpeded by GC: 269917
total cycles required for GC: 54230461

bus utilization: 55.27%
utilization due to cache invalidation requests: 0.91%
icache hit rate (288363469 hits / 288377613 fetches): 99.99%
of 14143 misses, average icache miss cost: 7.876

154
of 14143 Traditional misses, average icache miss cost: 7.876

dcache hit rate (80533197 hits / 92537036 fetches): 87.028%
of 1.20038e+07 misses, average dcache miss cost: 10.311
of 5.58549e+06 GC misses, average dcache miss cost: 18.459
of 526335 Traditional misses, average dcache miss cost: 15.219

Simulation completed on Sun Aug 2 17:12:56 1992
CPU usage: 120.1 system, 48535.1 user

4.4.3 lisp/db

dlxsimgc release:

Command line: slow/dlxsimgc.100000 -a lispHEAPUS __start

This simulation begun on Fri Jul 3 20:00:51 1992

Statistics:
Program image ends at 0x28610

| total machine instructions executed: 571680117 |
| cycles stalled for instruction fetch: 106270074 |
| cycles stalled for memory operations: 591281234 |
| cycles stalled following loads: 390129915 |
| cycles stalled for floating point results: 0 |
| cycles stalled for floating point processors: 0 |
| cycles stalled for branch-delay instruction fetches: 506626 |
| cycles stalled for trap interfacing: 36109 |

| total machine cycles executed: 1659904075 |
| total number of traps executed: 948 |

---

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>580</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>580</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>831260</td>
</tr>
<tr>
<td>While GC idle</td>
<td>3772296</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>831260</td>
</tr>
<tr>
<td>While GC idle</td>
<td>3772296</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
</tbody>
</table>

155
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
<th>Costs:</th>
<th>InitBlock</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>41031</td>
<td>87.122</td>
<td>49.7586</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>186707</td>
<td>86.2756</td>
<td>50.0808</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>41031</td>
<td>82.7612</td>
<td>49.1377</td>
<td>11</td>
</tr>
<tr>
<td>While GC idle</td>
<td>186707</td>
<td>81.9957</td>
<td>49.4082</td>
<td>8</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>7442</td>
<td>29.6922</td>
<td>14.3149</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>7442</td>
<td>22.7615</td>
<td>11.4918</td>
<td>3</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>18945</td>
<td>1680.31</td>
<td>2861.13</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>86990</td>
<td>24.9548</td>
<td>13.3786</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>18945</td>
<td>1672.61</td>
<td>2861.11</td>
<td>11</td>
</tr>
<tr>
<td>While GC idle</td>
<td>86990</td>
<td>15.9979</td>
<td>0.409794</td>
<td>5</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
Costs:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Latencies:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
<tbody>
<tr>
<td>stackPop</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Costs:
<table>
<thead>
<tr>
<th></th>
<th>During GC</th>
<th>While GC idle</th>
</tr>
</thead>
</table>

ScanBalance/Cycles | 6327170 | 0.555471 | 0.636969 | 0.000631114 | 475.545  
Heap Utilization   | 580      | 0.138924 | 0.101386152 | 0.128464 | 0.142387

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 4288858
total cycles required for GC: 531832677
bus utilization: 43.453%
utilization due to cache invalidation requests: 0.556%
icache hit rate (647103031 hits / 648028257 fetches): 99.857%
dcache hit rate (23198876 hits / 130436357 fetches): 17.786%

Simulation completed on Sun Jul 5 20:47:25 1992
CPU usage: 1026.7 system, 170862.2 user

4.4.4 lisp/prune

dlxsimgc release:

Command line: slow/dlxsimgc.80000 -a lispHEAPUS _start

This simulation begun on Wed Aug 5 20:01:15 1992

Statistics:
Program image ends at 0x28610

<table>
<thead>
<tr>
<th></th>
<th>329794042</th>
<th>61686998</th>
</tr>
</thead>
<tbody>
<tr>
<td>total machine instructions executed</td>
<td>319771088</td>
<td>227927653</td>
</tr>
<tr>
<td>cycles stalled for instruction fetch</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cycles stalled for memory operations</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cycles stalled following loads</td>
<td>2597</td>
<td>334772</td>
</tr>
<tr>
<td>cycles stalled for floating point results</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cycles stalled for floating point processors</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cycles stalled for branch-delay instruction fetches</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cycles stalled for trap interfacing</td>
<td>939463150</td>
<td>82</td>
</tr>
</tbody>
</table>

-------------------------------------------------------------------

157
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td></td>
<td>-----</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>675</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>675</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>501201</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2003498</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>501201</td>
</tr>
<tr>
<td>While GC idle</td>
<td>2003498</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>27917</td>
</tr>
<tr>
<td>While GC idle</td>
<td>112108</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>27917</td>
</tr>
<tr>
<td>While GC idle</td>
<td>112108</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>8700</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>8700</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>13939</td>
</tr>
<tr>
<td>While GC idle</td>
<td>55557</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>13939</td>
</tr>
<tr>
<td>While GC idle</td>
<td>55557</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>Alloc</td>
<td>Latencies</td>
</tr>
<tr>
<td>-------</td>
<td>-----------</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>AllocDSubSlice</td>
<td></td>
</tr>
<tr>
<td>AllocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>AllocDStack</td>
<td></td>
</tr>
<tr>
<td>AllocTStack</td>
<td></td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>ScanBalance/Cycles</td>
<td>3521697</td>
</tr>
<tr>
<td>Heap Utilization</td>
<td>674</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 2315153
total cycles required for GC: 339624914

bus utilization: 42.197%
utilization due to cache invalidation requests: 0.860%
icache hit rate (373066922 hits / 373676596 fetches): 99.837%
dcache hit rate (12714435 hits / 75571453 fetches): 16.824%

Simulation completed on Thu Aug 6 08:57:23 1992
CPU usage: 121.5 system, 46028.3 user
4.5 Experimental results for HC compiler

4.5.1 sfft/small

dlxsimgc release:

Command line: dlxsimgc.partinv.stall -a /usr1/schmidt/work/sfftHC
__start /usr1/schmidt/work/small

This simulation began on Wed Aug 12 19:19:26 1992

Statistics:
Program image ends at 0x33b40

total machine instructions executed: 61389042
cycles stalled for instruction fetch: 2857829
cycles stalled for memory operations: 24880225
cycles stalled following loads: 25431436
cycles stalled for floating point results: 1457403
cycles stalled for floating point processors: 16599
cycles stalled for branch-delay instruction fetches: 9094
cycles stalled for trap interfacing: 2558

-----------
total machine cycles executed: 116044186

-----------
total number of traps executed: 199

+-------------------------------------------------------------+------------------------------------------------------------+
| Arbiter | Operation Invocations                                      |
|---------+------------------------------------------------------------|
|         | Operations | number | mean*  | std dev | range |
|         |            |        |        |         |       |
|         | copyBlock  |        |        |         |       |
|         | Latencies:  |        |        |         |       |
|         | During GC | 0 | 0 | 0 |       |
|         | While GC idle | 345 | 78.1391 | 21.7483 |       |
|         | Costs:     |        |        |         |       |
|         | During GC | 0 | 0 | 0 |       |
|         | While GC idle | 345 | 70.3217 | 22.5603 | 35 | 91 |
|         | tendingDone|        |        |         |       |
|         | Latencies:  |        |        |         |       |
|         | During GC | 0 | 0 | 0 |       |
|         | While GC idle | 0 | 0 | 0 |       |
|         | Costs:     |        |        |         |       |
|         | During GC | 0 | 0 | 0 |       |
|         | While GC idle | 0 | 0 | 0 |       |
|         | allocInitRec|        |        |         |       |
|         | Latencies:  |        |        |         |       |
|         | During GC | 0 | 0 | 0 |       |
|         | While GC idle | 26 | 39 | 10.346 |       |
|         | Costs:     |        |        |         |       |
|         | During GC | 0 | 0 | 0 |       |
|         | While GC idle | 0 | 0 | 0 |       |
|         | copyPush   |        |        |         |       |
|         | Latencies:  |        |        |         |       |
|         | During GC | 0 | 0 | 0 |       |
|         | While GC idle | 0 | 0 | 0 |       |
|         | Costs:     |        |        |         |       |
|         | During GC | 0 | 0 | 0 |       |
|         | While GC idle | 0 | 0 | 0 |       |
|         | initBlock  |        |        |         |       |
|         | Latencies:  |        |        |         |       |
|         | During GC | 0 | 0 | 0 |       |


160
<table>
<thead>
<tr>
<th></th>
<th>350</th>
<th>109.969</th>
<th>12.8282</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>While GC idle</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idel</td>
<td>11</td>
<td>16.3636</td>
<td>0.504525</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idel</td>
<td>11</td>
<td>11</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>allocDSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>allocTSSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idel</td>
<td>0</td>
<td></td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>
During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0

stackPop

Latencies:

During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0

Costs:

During GC | 0 | 0 | 0 | 0 | 0 | 0
While GC idle | 0 | 0 | 0 | 0 | 0 | 0

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x834df4)

number of allocations unimpeded by GC: 37
total cycles required for GC: 0

bus utilization: 39.873%
utilization due to cache invalidation requests: 0.002%
icache hit rate (64064290 hits / 64079782 fetches): 99.976%
of 15491 misses, average icache miss cost: 7.054
of 15491 Traditional misses, average icache miss cost: 7.054
dcache hit rate (21743196 hits / 21818937 fetches): 99.653%
of 75741 misses, average dcache miss cost: 11.466
of 2435 GC misses, average dcache miss cost: 17.508
of 67896 Traditional misses, average dcache miss cost: 11.608

CPU usage: 238.2 system, 5732.5 user

4.5.2 lisp/prune
dlxsimgc release:

Command line: dlxsimgc.partinv.stall -a /usr1/schmidt/work/lispHC __start

This simulation begun on Wed Aug 12 20:13:16 1992

Statistics:
Program image ends at 0x2ba00

total machine instructions executed: 143619161
cycles stalled for instruction fetch: 46856628
cycles stalled for memory operations: 126150067
cycles stalled following loads: 81636880
cycles stalled for floating point results: 0
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 12538
cycles stalled for trap interfacing: 2094

---------
total machine cycles executed: 398277358

total number of traps executed: 82

Arbiter | Operation Invocations
-----------------------------------------------
Operations | number| mean*| std dev| range 162
<table>
<thead>
<tr>
<th>Operation</th>
<th>Latencies:</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tendingDone</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>2</td>
<td>-1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>2</td>
<td>-1</td>
<td>0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>119</td>
<td>693.899</td>
<td>1349.84</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>18671</td>
<td>-1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>119</td>
<td>688.899</td>
<td>1349.84</td>
<td>19</td>
<td>5646</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>18671</td>
<td>39.3316</td>
<td>8.3881</td>
<td>5</td>
<td>91</td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>initBlock</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>96</td>
<td>90.3021</td>
<td>37.1948</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>846</td>
<td>68.2778</td>
<td>44.7342</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>96</td>
<td>80.3854</td>
<td>38.28</td>
<td>13</td>
<td>124</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>846</td>
<td>57.0603</td>
<td>45.4631</td>
<td>8</td>
<td>124</td>
</tr>
<tr>
<td>tendDesc</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>26</td>
<td>28.5</td>
<td>10.5043</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>26</td>
<td>22.6923</td>
<td>11.8685</td>
<td>3</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>51</td>
<td>2060.96</td>
<td>4395.85</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>504</td>
<td>16.9702</td>
<td>1.97124</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>51</td>
<td>2055.96</td>
<td>4395.85</td>
<td>11</td>
<td>20685</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>504</td>
<td>11.9623</td>
<td>1.97312</td>
<td>11</td>
<td>16</td>
</tr>
<tr>
<td>allocDSlice</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocTSubSlice Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>allocTStack Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>stackPush Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>stackPop Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

ScrnBalance/Cycles: 11390 0.500151 0.463727 0.00194553 1
Heap Utilization: 2 0.216515 0.0141883 0.206892 0.226547

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 19302
total cycles required for GC: 1697336

bus utilization: 66.806%
utilization due to cache invalidation requests: 0.004%
icache hit rate (159537718 hits / 159553254 fetches): 99.99%
of 15535 misses, average icache miss cost: 10.415
of 15535 Traditional misses, average icache miss cost: 10.415
dcache hit rate (36928597 hits / 37143213 fetches): 99.42%
of 214616 misses, average dcache miss cost: 34.168
of 187097 GC misses, average dcache miss cost: 33.337
of 2679 Traditional misses, average dcache miss cost: 19.524

Simulation completed on Thu Aug 13 09:32:27 1992
CPU usage: 227.0 system, 16011.5 user
5 Raw Test Results for Chapter 7 Experiments

5.1 The RJS library

dlxsimgc release:

Command line: dlxsimgc.80000 rjsed __start

This simulation begun on Mon Aug 10 22:53:05 1992

Statistics:
Program image ends at 0x20d80

total machine instructions executed: 191481737
cycles stalled for instruction fetch: 49057934
cycles stalled for memory operations: 52401470
cycles stalled following loads: 36092683
cycles stalled for floating point results: 0
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 3011127
cycles stalled for trap interfacing: 1597138

please total machine cycles executed: 333642089

please total number of traps executed: 84066

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tenduringDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td>Operation</td>
<td>While GC idle</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>tcmEscape</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>allocLocal</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
<tr>
<td>allocCall</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
<tr>
<td>allocLocal</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
<tr>
<td>allocCall</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
<tr>
<td>allocLocal</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
<tr>
<td>allocCall</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Alloc Costs:</td>
<td></td>
</tr>
</tbody>
</table>
During GC  0|  0|  0|  0|  0|  0
While GC idle|  0|  0|  0|  0|  0|  0
stackPop  |  |  |  |  |  |  
Latencies:  |  |  |  |  |  |  
During GC  0|  0|  0|  0|  0|  0
While GC idle|  0|  0|  0|  0|  0|  0
Costs:  |  |  |  |  |  |  
During GC  0|  0|  0|  0|  0|  0
While GC idle|  0|  0|  0|  0|  0|  0

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x840000)

number of allocations unimpeded by GC:  0
total cycles required for GC:  0

bus utilization:  43.487%
utilization due to cache invalidation requests:  0.000%
icache hit rate ( 204614096 hits / 20716813 fetches):  98.766%
of 2.55572e+06 misses, average icache miss cost:  13.345
of 2.55572e+06 Traditional misses, average icache miss cost:  13.345
dcache hit rate ( 36416285 hits / 36532836 fetches):  99.681%
of 116551 misses, average dcache miss cost:  16.157
of 116551 Traditional misses, average dcache miss cost:  16.157

number of executions of 'malloc':  49440
total cycles for 'malloc' executions:  14101096
mean cycles per 'malloc' execution:  285.216
standard deviation of 'malloc' execution times:  521.384
range of malloc costs:  231 to 105858

number of executions of 'free':  268101
total cycles for 'free' executions:  27196075
mean cycles per 'free' execution:  101.440
standard deviation of 'free' execution times:  64.101
range of free costs:  59 to 1183

Simulation completed on Tue Aug 11 08:57:41 1992
CPU usage:  43.1 system, 16502.4 user

5.2  libg++

dlxsimgc release:

Command line: dlxsimgc.80000 libg++ __start

This simulation begun on Mon Aug 10 22:53:32 1992

Statistics:
   Program image ends at 0x3c260

   total machine instructions executed:  196643073
cycles stalled for instruction fetch:  53862716
cycles stalled for memory operations:  29227994
cycles stalled following loads:  39541578
cycles stalled for floating point results:  0
cycles stalled for floating point processors:  0
cycles stalled for branch-delay instruction fetches: 1977454
cycles stalled for trap interfacing: 1597058

---
total machine cycles executed: 322849863

total number of traps executed: 84158

---

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>copyBlock</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>tendingDone</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>allocInitRec</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>copyPush</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>initBlock</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>tendDesc</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>allocRec</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>allocDSlice</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Latencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

---
| While GC idle | 0 | 0 | 0 | 0 |
| Costs: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| allocTSlice |
| Latencies: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| Costs: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| allocDSSubSlice |
| Latencies: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| Costs: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| allocDStack |
| Latencies: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| Costs: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| allocTStack |
| Latencies: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| Costs: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| stackPush |
| Latencies: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| Costs: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| stackPop |
| Latencies: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |
| Costs: |
| During GC | 0 | 0 | 0 | 0 |
| While GC idle | 0 | 0 | 0 | 0 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x840000)

number of allocations unimpeded by GC: 0
total cycles required for GC: 0

bus utilization: 41.400%
utilization due to cache invalidation requests: 0.000%
icache hit rate (215486622 hits / 217168250 fetches): 99.226%
of 1.68163e+06 misses, average icache miss cost: 12.346
of 1.68163e+06 Traditional misses, average icache miss cost: 12.346
dcache hit rate (37641336 hits / 37763394 fetches): 99.677%
of 122058 misses, average dcache miss cost: 17.823
of 122058 Traditional misses, average dcache miss cost: 17.823

number of executions of 'malloc': 31369
total cycles for 'malloc' executions: 6886825
mean cycles per 'malloc' execution: 219.542
standard deviation of 'malloc' execution times: 80.162
range of malloc costs: 192 to 7978

number of executions of 'free': 31356
total cycles for 'free' executions: 3700065
mean cycles per 'free' execution: 118.002
standard deviation of 'free' execution times: 10.625
range of free costs: 93 to 915

Simulation completed on Tue Aug 11 08:58:19 1992
CPU usage: 43.1 system, 16534.5 user

5.3 The slice implementation

5.3.1 0x100000

dlxsimgc release:

Command line: dlxsimgc.100000.piv.st -a sliced.hc __start

This simulation begun on Sun Aug 16 18:41:41 1992

Statistics:
Program image ends at 0x20e50

total machine instructions executed: 112949857
cycles stalled for instruction fetch: 44864860
cycles stalled for memory operations: 62112240
cycles stalled following loads: 96830952
cycles stalled for floating point results: 0
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 3486182
cycles stalled for trap interfacing: 2231313

----------
total machine cycles executed: 322478404

----------
total number of traps executed: 64035

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>GC Latencies</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>allocInitRec</td>
<td>During GC: 37</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC: 37</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC: 0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC: 0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC: 44</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC: 44</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendDesc</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC: 16</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC: 16</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocRec</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC: 6</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC: 6</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDSSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC: 0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC: 0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocTSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC: 73</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC: 73</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td>Latencies:</td>
</tr>
<tr>
<td></td>
<td>During GC: 0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td></td>
<td>Costs:</td>
</tr>
<tr>
<td></td>
<td>During GC: 0</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>11</td>
</tr>
<tr>
<td>While GC idle</td>
<td>9549</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>11</td>
</tr>
<tr>
<td>While GC idle</td>
<td>9549</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>ScanBalance/Cycles</td>
<td>17727</td>
</tr>
<tr>
<td>Heap Utilization</td>
<td>2</td>
</tr>
</tbody>
</table>

* A value of -1 signifies that the mean is not available due to unclaimed results.

---

number of allocations unimpeded by GC: 30281
total cycles required for GC: 2000729

bus utilization: 67.043%
utilization due to cache invalidation requests: 0.005%
icache hit rate (17962429 hits / 12160817 fetches): 96.976%
of 3.6783e+06 hits, average icache miss cost: 9.108
of 3.67839e+06 Traditional misses, average icache miss cost: 9.108
dcache hit rate (27887234 hits / 28218945 fetches): 98.825%
of 331711 misses, average dcache miss cost: 33.792
of 293729 GC misses, average dcache miss cost: 34.092
of 500 Traditional misses, average dcache miss cost: 27.168

Simulation completed on Sun Aug 16 22:35:23 1992
CPU usage: 47.10 system, 12629.5 user

5.3.2 0x2000000
dlxsimgc release:
Command line: dlxsimgc.200000.piv.st -a sliced.hc __start

This simulation begun on Sun Aug 16 22:53:31 1992

Statistics:
  Program image ends at 0x20e50

  total machine instructions executed:     112947888
  cycles stalled for instruction fetch:    38769701
  cycles stalled for memory operations:    49081843
  cycles stalled following loads:          76990957
  cycles stalled for floating point results: 0
  cycles stalled for floating point processors: 0
  cycles stalled for branch-delay instruction fetches: 2614267
  cycles stalled for trap interfacing:      1947820

  total machine cycles executed:          282252476
  total number of traps executed:         64035

<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>------------------</td>
<td>--------</td>
</tr>
<tr>
<td>copyBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendingDone</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>allocInitRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>copyPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>initBlock</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td></td>
<td>While GC idle</td>
</tr>
<tr>
<td>tendDesc</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>During GC</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>8</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocRec</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>3</td>
</tr>
<tr>
<td>While GC idle</td>
<td>19</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>3</td>
</tr>
<tr>
<td>While GC idle</td>
<td>19</td>
</tr>
<tr>
<td>allocDSSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>47</td>
</tr>
<tr>
<td>While GC idle</td>
<td>12440</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>47</td>
</tr>
<tr>
<td>While GC idle</td>
<td>12440</td>
</tr>
<tr>
<td>allocDSSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>9560</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>9560</td>
</tr>
<tr>
<td>allocDStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>allocTStack</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPush</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>stackPop</td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td></td>
</tr>
<tr>
<td>During GC</td>
<td>0</td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
</tr>
<tr>
<td>Costs:</td>
<td></td>
</tr>
</tbody>
</table>
During GC| 0| 0| 0| 0| 0| 0
While GC idle| 0| 0| 0| 0| 0| 0

ScanBalance/Cycles| 13924| 0.771877| 0.957836| 0.00290698| 103.63
Heap Utilization| 1| 0.0504303| 0| 0.0504303| 0.0504303

* A value of -1 signifies that the mean is not available due to unclaimed results.

number of allocations unimpeded by GC: 30256
total cycles required for GC: 150191

bus utilization: 61.35%
utilization due to cache invalidation requests: 0.003%
icache hit rate (117960124 hits / 121638404 fetches): 96.976%
of 3.67828e+06 misses, average icache miss cost: 8.160
dcache hit rate (2788884 hits / 2821654 fetches): 98.829%
of 330470 misses, average dcache miss cost: 26.077
of 292833 GC misses, average dcache miss cost: 26.135
of 454 Traditional misses, average dcache miss cost: 18.659

Simulation completed on Mon Aug 17 02:57:38 1992
CPU usage: 23.7 system, 14198.7 user

5.3.3 0x400000

dlxsimc release:

Command line: dlxsimc.400000.piv.st -a sliced.hc __start

This simulation begun on Mon Aug 17 09:06:02 1992

Statistics:
Program image ends at 0x20e50

total machine instructions executed: 112945947
cycles stalled for instruction fetch: 34700531
cycles stalled for memory operations: 40889165
cycles stalled following loads: 64544427
cycles stalled for floating point results: 0
cycles stalled for floating point processors: 0
cycles stalled for branch-delay instruction fetches: 2053715
cycles stalled for trap interfacing: 1593201

total machine cycles executed: 256726986

total number of traps executed: 64035

---------------------------------------------------------------
<table>
<thead>
<tr>
<th>Arbiter</th>
<th>Operation Invocations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>number</td>
</tr>
<tr>
<td>----------------------------------------------------------</td>
<td></td>
</tr>
</tbody>
</table>
copyBlock | | | | |
Latencies: | | | | |
| During GC | 0| 0| 0| 0 |
| While GC idle| 0| 0| 0| 0 |
Costs: | | | | |
| During GC | 0| 0| 0| 0 |

175
<table>
<thead>
<tr>
<th></th>
<th>tendingDone</th>
<th>Latencies:</th>
<th>Costs:</th>
<th>allocInitRec</th>
<th>Latencies:</th>
<th>Costs:</th>
<th>allocRec</th>
<th>Latencies:</th>
<th>Costs:</th>
<th>allocDSlice</th>
<th>Latencies:</th>
<th>Costs:</th>
<th>allocTSlice</th>
<th>Latencies:</th>
<th>Costs:</th>
<th>allocDSubSlice</th>
<th>Latencies:</th>
<th>Costs:</th>
</tr>
</thead>
<tbody>
<tr>
<td>While GC idle</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8163</td>
<td>24.13</td>
<td>5.51047</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>costs:</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>752</td>
<td>110.246</td>
<td>12.4514</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>initBlock</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>752</td>
<td>99.6915</td>
<td>12.2945</td>
<td>8</td>
<td>113</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>tendDesc</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>18</td>
<td>16.2222</td>
<td>0.427793</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>18</td>
<td>11</td>
<td>0</td>
<td>11</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>allocDSlice</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>12486</td>
<td>26.2628</td>
<td>6.18736</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>12486</td>
<td>21.2628</td>
<td>6.18736</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>allocDSubSlice</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>12486</td>
<td>21.2628</td>
<td>6.18736</td>
<td>13</td>
<td>38</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>12486</td>
<td>21.2628</td>
<td>6.18736</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>allocTSlice</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Latencies:</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costs:</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>While GC idle</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>allocTSubSlice</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Latencies: |
| During GC | 0 | 0 | 0 |
| While GC idle | 9560 | 26.0002 | 0.0204551 |
|
Costs: |
| During GC | 0 | 0 | 0 |
| While GC idle | 9560 | 21.0002 | 0.0204551 |
|
allocDStack |
Latencies: |
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 |
|
Costs: |
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 |
|
stackPush |
Latencies: |
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 |
|
Costs: |
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 |
|
stackPop |
Latencies: |
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 |
|
Costs: |
| During GC | 0 | 0 | 0 |
| While GC idle | 0 | 0 |

* A value of -1 signifies that the mean is not available due to unclaimed results.

warning: no garbage collection activity (New = 0x8d6e40)

| number of allocations unimpeded by GC: | 30227 |
| total cycles required for GC: | 0 |

| bus utilization: | 56.75% |
| utilization due to cache invalidation requests: | 0.000% |
| icache hit rate (117957945 hits / 121636026 fetches): | 96.976% |
| of 3.67808e+06 misses, average icache miss cost: | 7.533 |
| of 3.67808e+06 Traditional misses, average icache miss cost: | 7.533 |
| dcache hit rate (27888210 hits / 28218370 fetches): | 98.834% |
| of 329160 misses, average dcache miss cost: | 21.224 |
| of 398723 GC misses, average dcache miss cost: | 21.212 |
| of 396 Traditional misses, average dcache miss cost: | 12.429 |

Simulation completed on Mon Aug 17 13:33:56 1992
CPU usage: 41.7 system, 13659.1 user

References

