

Experimental Assessment of the Perfect Club Benchmarks on a Cray Y-MP

K. A. Robbins S. Robbins

April, 1992

Division of Mathematics, Computer Science, and Statistics
The University of Texas at San Antonio
San Antonio, TX 78249

krobbins@runner.utsa.edu
512-691-5543

srobbins@runner.utsa.edu
512-691-5544

Abstract

This study provides an experimental assessment of the dynamic behavior of the Perfect Club Benchmarks for the memory behavior as run on a Cray Y-MP. The goal of this study is to provide a basis for the generation of realistic synthetic memory workloads and port utilization estimates for shared memory machines. An analysis of memory traces of the Perfect Club Benchmarks run on a Cray Y-MP processor is presented. Port utilization is shown to be surprisingly low, and the dynamic behavior of memory reference streams for the programs varied considerably over the duration of execution. All of the programs evolved in distinct phases after a relatively short initial phase.

Key words: benchmarks, Perfect Club, Cray Y-MP, memory utilization, shared memory.

University of Texas at San Antonio Technical Report UTSA-CS-92-102

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 1 |
| 2 | Methodology | 2 |
| 3 | Results of the measurements | 4 |
| 3.1 | Program phases | 5 |
| 3.2 | Granularity | 6 |
| 3.3 | Asymmetry in port utilization | 6 |
| 3.4 | Scalar clustering | 7 |
| 4 | Conclusions | 8 |
| A | Hardware Performance Monitor Data | 50 |
| A.1 | Statistics for ADM run on hpm | 51 |
| A.2 | Statistics for ARC2D run on hpm | 52 |
| A.3 | Statistics for BDNA run on hpm | 53 |
| A.4 | Statistics for DYFESM run on hpm | 54 |
| A.5 | Statistics for FLO52 run on hpm | 55 |
| A.6 | Statistics for MDG run on hpm | 56 |
| A.7 | Statistics for MG3D run on hpm | 57 |
| A.8 | Statistics for OCEAN run on hpm | 58 |
| A.9 | Statistics for QCD run on hpm | 59 |
| A.10 | Statistics for SPEC77 run on hpm | 60 |
| A.11 | Statistics for SPICE run on hpm | 61 |
| A.12 | Statistics for TRACK run on hpm | 62 |
| A.13 | Statistics for TRFD run on hpm | 63 |
| B | Data from sim | 64 |
| B.1 | Statistics for ADM run under sim | 65 |
| B.2 | Statistics for ARC2D run under sim | 67 |
| B.3 | Statistics for BDNA run under sim | 69 |
| B.4 | Statistics for DYFESM run under sim | 71 |

| | | |
|---------------------|---|-----------|
| B.5 | Statistics for FLO52 run under sim | 73 |
| B.6 | Statistics for MDG run under sim | 75 |
| B.7 | Statistics for MG3D run under sim | 77 |
| B.8 | Statistics for OCEAN run under sim | 79 |
| B.9 | Statistics for QCD run under sim | 81 |
| B.10 | Statistics for SPEC77 run under sim | 83 |
| B.11 | Statistics for SPICE run under sim | 85 |
| B.12 | Statistics for TRACK run under sim | 87 |
| B.13 | Statistics for TRFD run under sim | 89 |
| Bibliography | | 91 |

List of Figures

| | | |
|----|---|----|
| 1 | Port utilization for Perfect Benchmark ADM. | 9 |
| 2 | Port utilization histogram for Perfect Benchmark ADM. | 10 |
| 3 | Distributions of lengths and strides for Perfect Benchmark ADM. . . | 11 |
| 4 | Port utilization for Perfect Benchmark ARC2D. | 12 |
| 5 | Port utilization histogram for Perfect Benchmark ARC2D. | 13 |
| 6 | Distribution of lengths and strides for Perfect Benchmark ARC2D. . . | 14 |
| 7 | Port utilization for Perfect Benchmark BDNA. | 15 |
| 8 | Port utilization histogram for Perfect Benchmark BDNA. | 16 |
| 9 | Distribution of lengths and strides for Perfect Benchmark BDNA . . . | 17 |
| 10 | Port utilization for Perfect Benchmark DYFESM | 18 |
| 11 | Port utilization histogram for Perfect Benchmark DYFESM | 19 |
| 12 | Distribution of lengths and strides for Perfect Benchmark DYFESM. | 20 |
| 13 | Port utilization for Perfect Benchmark FLO52. | 21 |
| 14 | Port utilization histogram for Perfect Benchmark FLO52. | 22 |
| 15 | Distribution of lengths and strides for Perfect Benchmark FLO52. . . | 23 |
| 16 | Port utilization for Perfect Benchmark MDG. | 24 |
| 17 | Port utilization histogram for Perfect Benchmark MDG. | 25 |
| 18 | Distribution of lengths and strides for Perfect Benchmark MDG. . . | 26 |
| 19 | Port utilization for Perfect Benchmark MG3D. | 27 |
| 20 | Port utilization histogram for Perfect Benchmark MG3D. | 28 |
| 21 | for Perfect Benchmark MG3D. | 29 |
| 22 | Port utilization for Perfect Benchmark OCEAN. | 30 |
| 23 | Port utilization histogram for Perfect Benchmark OCEAN. | 31 |
| 24 | Distribution of lengths and strides for Perfect Benchmark OCEAN. . . | 32 |
| 25 | Port utilization for Perfect Benchmark QCD. | 33 |
| 26 | Port utilization histogram for Perfect Benchmark QCD. | 34 |
| 27 | Distribution of lengths and strides for Perfect Benchmark QCD. . . . | 35 |
| 28 | Port utilization for Perfect Benchmark SPEC77. | 36 |
| 29 | Port utilization histogram for Perfect Benchmark SPEC77. | 37 |
| 30 | Distribution of lengths and strides for Perfect Benchmark SPEC77. . . | 38 |

| | | |
|----|---|----|
| 31 | Port utilization for Perfect Benchmark SPICE. | 39 |
| 32 | Port utilization histogram for Perfect Benchmark SPICE. | 40 |
| 33 | Distribution of lengths and strides for Perfect Benchmark SPICE. . . | 41 |
| 34 | Port utilization for Perfect Benchmark TRACK. | 42 |
| 35 | Port utilization histogram for Perfect Benchmark TRACK. | 43 |
| 36 | Distribution of lengths and strides for Perfect Benchmark TRACK. . | 44 |
| 37 | Port utilization for Perfect Benchmark TRFD. | 45 |
| 38 | Port utilization for Perfect Benchmark TRFD. with a window size of 100,000 | 46 |
| 39 | Port utilization for Perfect Benchmark TRFD. with a window size of 10,000 | 47 |
| 40 | Port utilization histogram for Perfect Benchmark TRFD. | 48 |
| 41 | Distributions of lengths and strides for Perfect Benchmark TRFD. . | 49 |

1 Introduction

This technical report describes the memory access behavior of the Perfect Club Benchmarks [2] on a Cray Y-MP. These programs, which have been drawn from scientific and engineering applications, have been developed in a cooperative effort by several universities, research centers, and industry. A number of interesting observations are drawn from the data. The study was motivated by the need for a better understanding of memory utilization in real programs. It is hoped that this understanding will allow memory designers to devise more accurate synthetic workloads for testing new memory designs.

Memory utilization of shared memory machines under normal circumstances is surprisingly low as shown by the data in Table 1. Table 1 presents 7 months of hardware performance monitor data for a Cray Y-MP system at NCSA[3]. This data was obtained by a kernel modification which caused the group 0 hardware performance monitor counters to be saved after each job. The data was filtered to remove short jobs. The column labeled *Aver port ref rate* is related to the actual port utilization, but it does not take into account the setup and shutdown times for vector references, the effect of holds, or the asymmetry in the three Cray Y-MP memory ports. It also counts only one cycle for a scalar access. The maximum reference rate which can be supported by a Cray Y-MP occurs if all three ports are performing vector operations of maximal length (64). This rate is calculated by assuming that vector reads have a combined setup and shutdown time of 5 cycles while writes have a combined setup and shutdown time of 6 cycles. Since there are two read ports and one write port the maximum processor reference rate should be:

$$2\frac{64}{69} + \frac{64}{70} = 2.769 \text{ references per cycle}$$

The data showed that the maximum reference rate achieved by any program run during those 7 months was 2.04 references per cycle. The average reference rate varied from 0.292 in October 1991 to 0.346 in December 1991.

| Month | Programs | Total cycles | Max ref per cycle | Aver ref per cycle | Aver port ref rate |
|-----------|----------|------------------------|----------------------|-----------------------|-----------------------|
| June | 15, 222 | 4.97×10^{14} | 1.93 | 0.313 | 0.104 |
| July | 21, 899 | 8.34×10^{14} | 1.51 | 0.312 | 0.104 |
| August | 12, 710 | 4.94×10^{14} | 1.88 | 0.325 | 0.108 |
| September | 28, 302 | 7.11×10^{14} | 1.58 | 0.339 | 0.113 |
| October | 25, 584 | 8.34×10^{14} | 2.03 | 0.292 | 0.097 |
| November | 50, 623 | 9.25×10^{14} | 1.43 | 0.351 | 0.117 |
| December | 23, 756 | 11.04×10^{14} | 2.04 | 0.346 | 0.115 |

Table 1: Group 0 hpm data for last half of 1991 at NCSA.

The low reference rate in this data suggested that a more detailed analysis was necessary in order to understand the utilization statistics. The Perfect Club Benchmark Suite was selected because of the variety of program behaviors present in relatively short programs. In the next section, a description of the measurement methodology is presented and compared with techniques used by other researchers. In section 3 a summary of the results for the Perfect Club Benchmarks is given. Finally conclusions about the techniques used in this paper are discussed. Appendices A and B contain a more detailed summary of the data.

2 Methodology

Two studies of the Perfect Club Benchmarks on Cray systems have recently appeared. Malony, Larson, and Reed [4] have developed a tracing facility for the Cray X-MP and Cray 2 in which they use Cray compiler support for automatic instrumentation to dump and analyze hardware performance monitor (hpm) data on exit and entry to subroutines. This is a general, relatively low-cost technique for obtaining dynamic information about program behavior. It uses the real time clock for timing and is therefore susceptible to errors due to multiprogramming. The technique also introduces some overhead, which in some cases can be analyzed and removed.

Vajapeyam, Sohi, and Hsu [7] decomposed programs into basic blocks and used the Cray tool `jumptrace` to obtain execution frequencies of the basic blocks. They also obtained hardware performance monitor data for the entire programs. This technique is also a relatively fast, low overhead method of obtaining information about program execution. Their technique does not provide dynamic information

about program execution. In addition, they did not profile library routines in the basic blocks, and since the vector length was not available, they used average vector lengths to estimate utilization.

Neither of the above studies was designed to provide information about the utilization of the individual ports and therefore could not be used to provide accurate synthetic workload models.

This study uses a memory reference tracing facility in `sim` which became available with release 6.0 of the UNICOS operating system. `sim` is a Cray Y-MP simulator which provides exact, instruction level timing of programs run on a single Cray Y-MP processor ignoring memory conflicts. `sim` takes a fully-linked object module as input and traces execution of the program. The memory tracing facility generates a file which has an entry containing the opcode, the time of issue, the starting address, the stride, and the vector length for each memory reference. For scalar operations the vector lengths and strides are 1. This file was then used as input to a program which profiled port utilization.

This approach has two advantages: it is nonintrusive and it allows the study of memory reference patterns with arbitrarily fine granularity. Unfortunately a program run under `sim` on a Cray Y-MP executes more than 1000 times slower than it would normally take. The memory reference trace files generated by `sim` are also very large. The shorter traces were 200 to 500 megabytes in compressed format. The larger benchmarks had to be checkpointed and run in pieces to produce files that could fit on a single disk. As mentioned above, `sim` assumes that there are no memory conflicts, so it gives slightly different timing values than provided directly by `hpm`.

The programs were run under version 6.14 of UNICOS on a Cray Y-MP8/864. The Perfect Club Benchmark Suite 1 was used for the baseline runs. Minor modifications had to be made to the source in order to allow compilation under version 4.0.4.4 of the Cray Fortran compiler.

The port utilization program processed the trace information produced by `sim` and kept track of the number of cycles in a given time partition (10,000 cycles for this paper) during which each port was busy. It also kept a record of the distribution of strides and vector lengths for each of the programs. It used the Cray Y-MP port

assignment strategy as now described. All writes (scalar, block, and vector) are performed on port C. Vector and block reads are performed on ports A and B. The default port for such reads is port B. Scalar and vector operations are not allowed to proceed simultaneously. An estimate was kept of the number of possible cycles in which a vector reference was delayed because a scalar operation was in progress and vice versa. (This upper bound was calculated by counting the number of times that a vector reference was initiated on the cycle after a scalar reference completed.) The sum of the setup and shutdown times for vector reads is 5 cycles and for vector writes is 6 cycles, and this time was counted in the port utilization. Scalar write operations use port C for 5 cycles, but consecutive scalar operations can be done on consecutive cycles. Similarly, scalar read operations use port A for 5 cycles, but consecutive scalar reads can be done on consecutive cycles.

3 Results of the measurements

A summary of the hardware performance monitor measurements for the Perfect Club Benchmarks is given in Table 2. The complete hardware performance monitor data is given in Appendix A. The `hpm` gives a count of the number of memory conflicts. In one of the benchmarks, SPEC77, there were more memory conflicts than actual memory references. Note that the exact number of conflicts depends on how other processes on the system are accessing memory during the run. The percentage of reads for the entire Perfect Club suite varied from 58% to 71% which is consistent with the commonly assumed 2/3 fraction of reads.

| Program | Cycles | Instructions | | Memory | | % Memory | | % Vect |
|---------|--------|--------------|--------|--------|-------|----------|-------|--------|
| | | total | memory | refs | confs | confs | reads | refs |
| ADM | 3916 | 1415 | 164 | 693 | 223 | 32.09 | 60.67 | 84.33 |
| ARC2D | 2212 | 400 | 40 | 1937 | 106 | 5.47 | 70.82 | 99.88 |
| BDNA | 1534 | 301 | 34 | 669 | 42 | 6.35 | 58.63 | 96.89 |
| DYFESM | 2029 | 580 | 50 | 833 | 49 | 5.91 | 57.69 | 97.71 |
| FLO52 | 902 | 239 | 23 | 717 | 38 | 5.29 | 71.47 | 99.21 |
| MDG | 28824 | 8521 | 259 | 2842 | 186 | 6.55 | 61.59 | 62.04 |
| MG3D | 20843 | 6057 | 749 | 10370 | 1601 | 15.44 | 58.22 | 96.17 |
| OCEAN | 3482 | 960 | 216 | 2344 | 223 | 9.53 | 51.39 | 92.68 |
| QCD | 3295 | 1234 | 137 | 323 | 18 | 5.42 | 58.04 | 58.89 |
| SPICE | 1363 | 327 | 88 | 101 | 11 | 11.35 | 71.02 | 15.40 |
| SPEC77 | 9010 | 2898 | 240 | 1689 | 2045 | 121.10 | 70.01 | 93.99 |
| TRACK | 1696 | 497 | 70 | 124 | 6 | 4.70 | 65.66 | 45.35 |
| TRFD | 1321 | 525 | 42 | 670 | 248 | 37.07 | 66.03 | 98.13 |

Table 2: Hardware performance monitor data for the Perfect Club benchmarks. All counts are in millions.

Following [7] TRACK, SPICE, and QCD are classified as scalar code, while BDNA, MG3D, FLO52, ARC2D, SPEC77, and MDG are classified as vector code. The remaining benchmarks are assumed to be mixtures by Vajapeyam et al, but are almost completely vectorized as far as memory references are concerned.

The results of dynamic memory behavior as obtained from `sim` are discussed in the remainder of this report. A summary of statistics generated for the `sim` runs is given in Appendix B. Figure 1 through 40 show the memory behavior for the 13 Perfect Club Benchmarks. There are four noteworthy aspects to the memory references patterns which are presented in the following subsections: phases, granularity, asymmetry in port utilization, and scalar clustering.

3.1 Program phases

All of the programs had start-up phases which were relatively short compared to the overall program duration. Many of the programs evolved through distinct phases with unique patterns of memory activity. Figure 13 shows the memory activity for the three memory ports during the execution of the FLO52 vector benchmark. Each point on the graph represents an average of 1,000,000 machine cycles. The average utilization for ports A, B, and C are 0.275, 0.391, and 0.279 respectively. Since

99% of the memory operations are vector operations, the percentage of reads can be estimated from the port utilization by the sum of the utilizations of A and B divided by the total utilization. This estimate gives the percentage of reads as 70.9% which is good agreement with the value 71.47% obtained directly from the hardware performance monitor. FLO52 has very short startup and shutdown phases. The main computation has three distinct phases which have been noted in [4] to correspond to three distinct grid calculations.

Scalar code tends to show substantially less phase behavior. Port utilization for the scalar program SPICE is shown in Figure 31. As expected, port B is virtually quiescent while ports A and C are relatively more active with port A (reads) about twice as active as port C (writes). The overall memory utilization is quite low and fairly uniform over the duration of the program.

3.2 Granularity

Each program has a characteristic granularity and different aspects of utilization substructure appear when averages over shorter windows are taken. Figure 37 shows the port utilization for the benchmark TRFD when activity is averaged over 1,000,000 cycles. The phase activity is shown clearly in both averages, but additional substructure appears in the shorter averages. In Figure 38 a blowup of two regions of the window size 100,000 plot shows more details. In Figure 39 a window size of 10,000 is used and the structure of the part of the program which is equivalent to window 6500 in the window size 100,000 plot is shown. This graph illustrates very regular memory reference activity with a wide degree of local variation occurring during loops.

3.3 Asymmetry in port utilization

As mentioned above scalar code results in a very asymmetric utilization of the ports, since scalar references can only be made on ports A and C. Vector code utilizes the ports in a roughly equal manner, but port B always has a higher port utilization than port A because of the priority scheme used in assigning vector references. The 2/3 rule for percentage of reads implies that the utilization of port C should be the average of ports A and B. In general if A , B , and C represent the utilizations (fractions of

cycles that the port is busy) of their respective ports and x is the fraction of reads, then the following relationship should hold:

$$x = \frac{A + B}{A + B + C}$$

Table 3 gives the predicted and computed x for the benchmarks.

| Benchmark | A | B | C | x | x pred | % vector |
|-----------|-------|-------|-------|-------|----------|-------------|
| ADM | 0.130 | 0.094 | 0.146 | 0.607 | 0.605 | 84.33 |
| ARC2D | 0.269 | 0.437 | 0.296 | 0.708 | 0.705 | 99.88 |
| BDNA | 0.140 | 0.185 | 0.226 | 0.586 | 0.590 | 96.89 |
| DYFESM | 0.157 | 0.176 | 0.222 | 0.577 | 0.600 | 97.71 |
| FLO52 | 0.296 | 0.391 | 0.268 | 0.725 | 0.720 | 99.21 |
| MDG | 0.104 | 0.048 | 0.098 | 0.616 | 0.608 | 62.04 |
| MG3D | 0.172 | 0.221 | 0.291 | 0.582 | 0.575 | 96.17 |
| OCEAN | 0.229 | 0.205 | 0.453 | 0.514 | 0.489 | 92.68 |
| QCD | 0.136 | 0.026 | 0.092 | 0.580 | 0.637 | 58.89 |
| SPEC77 | 0.112 | 0.138 | 0.117 | 0.700 | 0.681 | 93.99 |
| SPICE | 0.194 | 0.011 | 0.082 | 0.710 | 0.714 | 15.40 |
| TRACK | 0.138 | 0.022 | 0.074 | 0.657 | 0.684 | 45.35 |
| TRFD | 0.192 | 0.304 | 0.266 | 0.660 | 0.651 | 98.13 |

Table 3: Port utilization and fraction of reads for vectorized code.

3.4 Scalar clustering

In producing a model for generating realistic synthetic workloads, it is useful to look at the probability distributions for the fraction of cycles busy. The probability distributions were computed by calculating the histogram of the fraction of cycles busy in each 10,000 cycle partition. A resolution of 100 bins (0.01) was used for the histograms. These distributions have a markedly different form for vector and scalar code. Scalar code is typified by the behavior of SPICE as shown in Figure 32. Port B has a roughly exponential distribution which decays rapidly while ports A and C have distributions which are peaked around their mean. There is relatively little spreading.

The situation for vector code is more complicated. The probability distributions tend to be broader. Figure 40 shows the probability distributions for the three ports

for the benchmark TRFD. Port B shows greater activity and a higher variance than ports A or C.

4 Conclusions

This report presented an experimental study of memory and port utilization obtained from the Perfect Club Benchmarks in order to provide a basis for realistic modeling of memory utilization. While the technique used for this study was extremely costly in computer time, it has provided a basis for simple modeling of memory and port utilization [6] which can be obtained directly from hardware performance monitor data. Synthetic memory models for scalar references should allow for the asymmetry of port utilization and should assume a distribution of time between references rather than a fixed probability of a reference occurring. A reasonable model for vector references assumes a relatively broad and fairly uniform distribution centered around the average utilization rate.

Acknowledgments:

The authors would like to thank John Larson of CSRD at University of Illinois for his helpful comments. Doru Marcusiu at NCSA made the data available for Table 1. Greg Faanes of Cray Research provided help with `sim`. Charles Grassl of Cray Research provided optimized versions of the Perfect Club Benchmarks and useful advice on compilation. This work was partially supported by Cray Research, the University of Texas System for High Performance Computing, and the University of Texas at San Antonio Faculty Research Awards program.

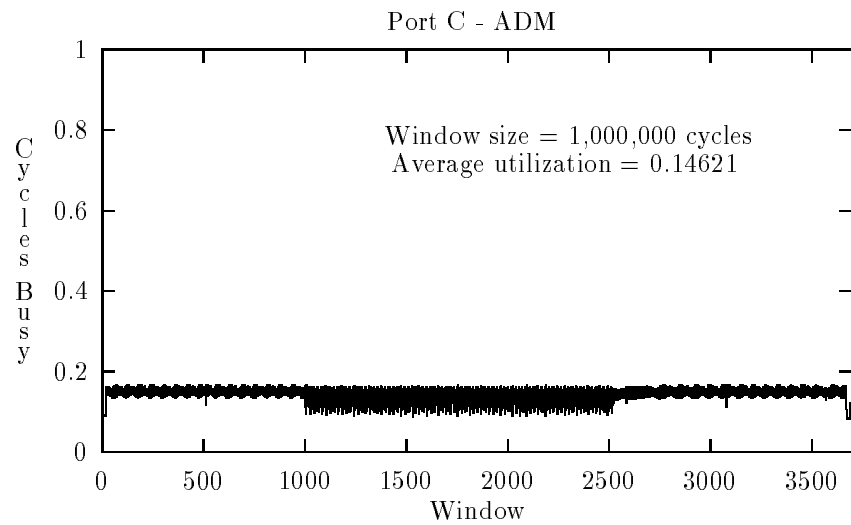
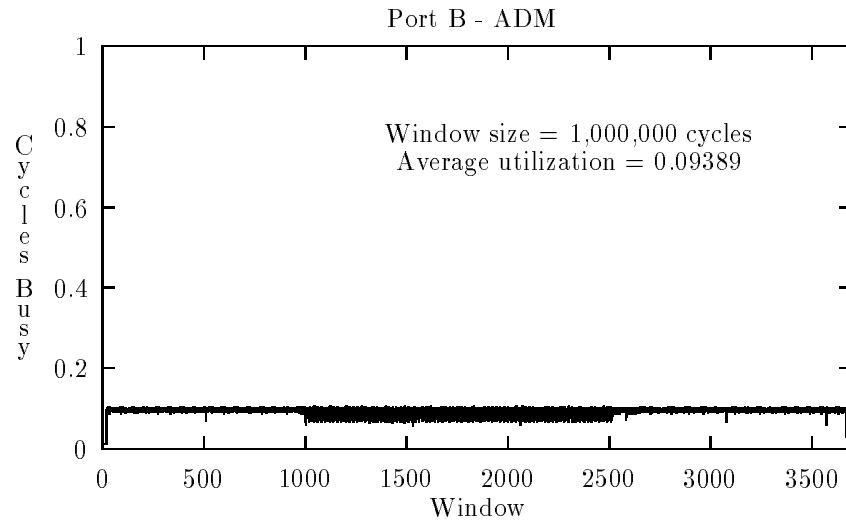
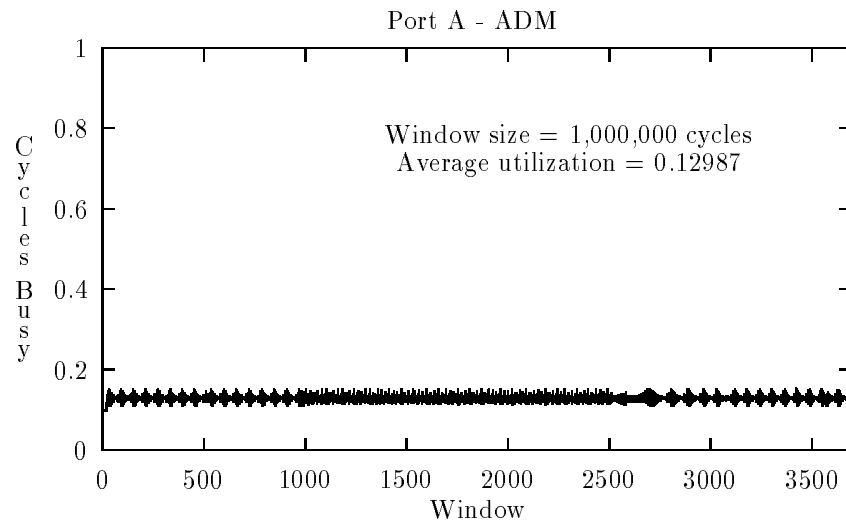


Figure 1: Port utilization for Perfect Benchmark ADM.

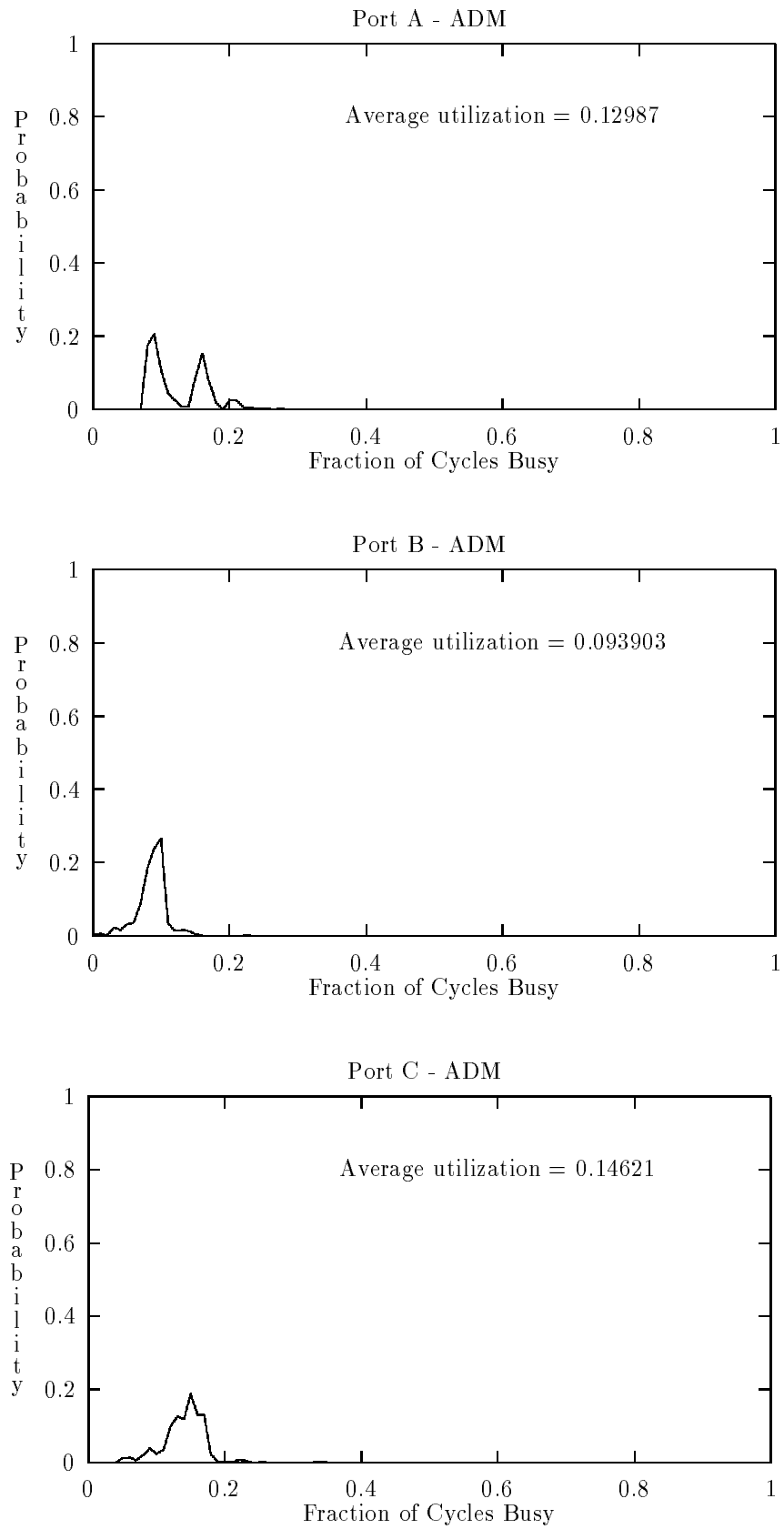


Figure 2: Port utilization histogram for Perfect Benchmark ADM.

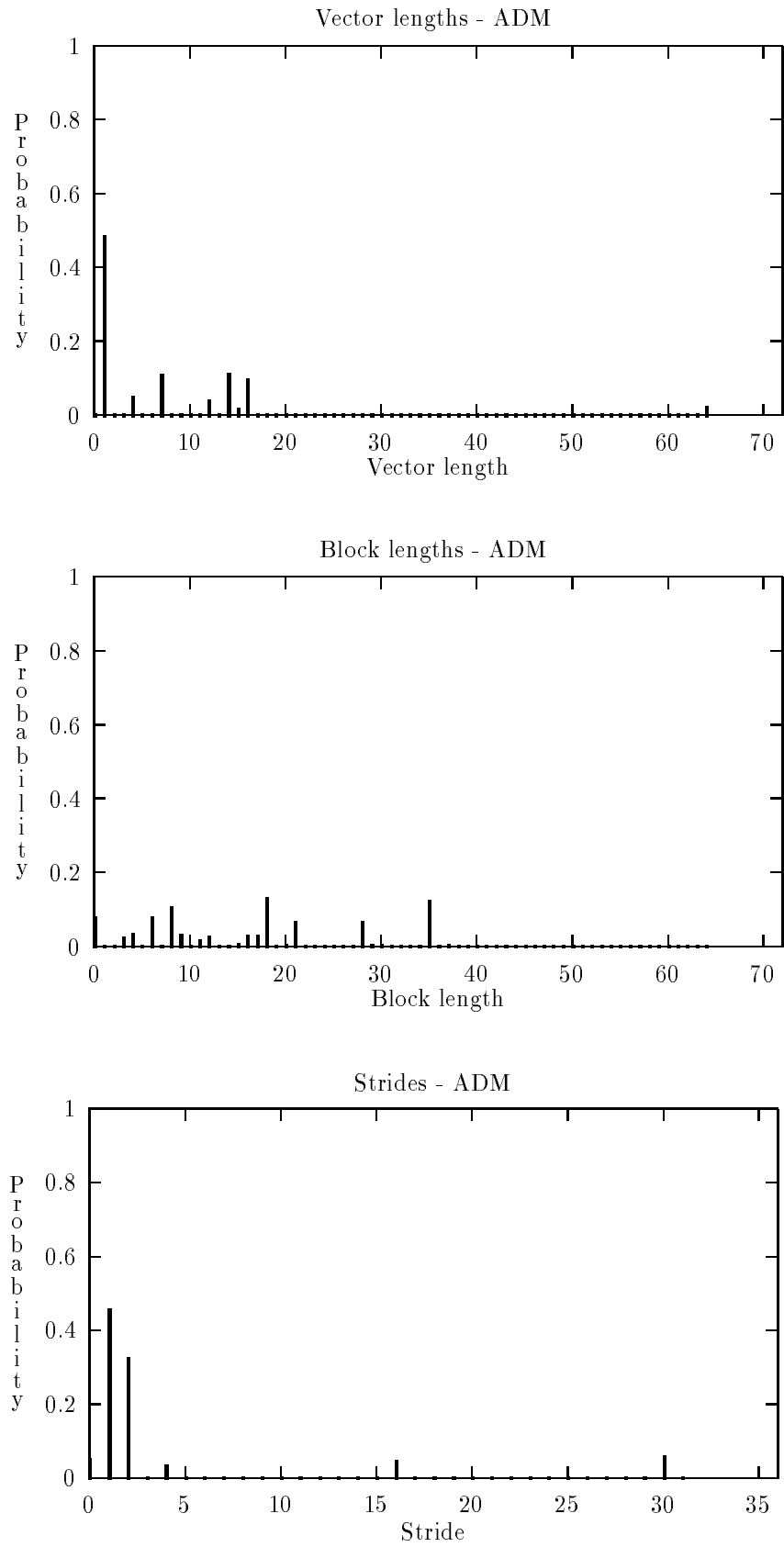


Figure 3: Distributions of lengths and strides for Perfect Benchmark ADM.

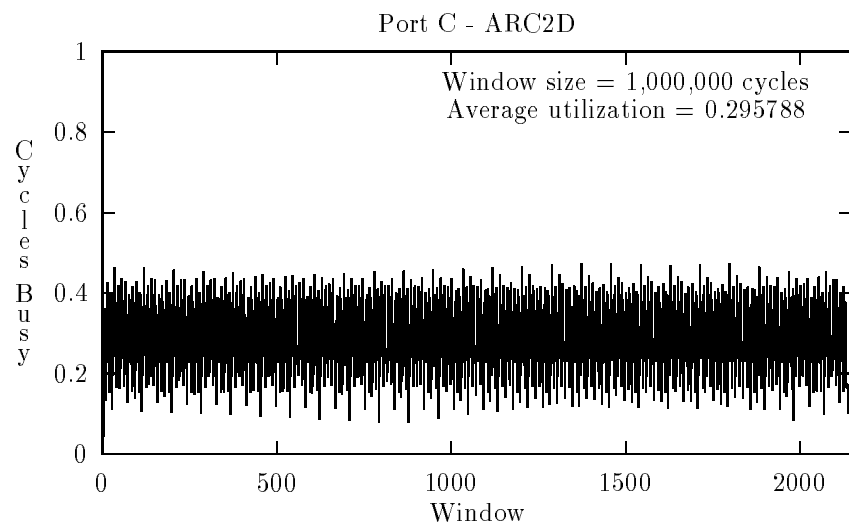
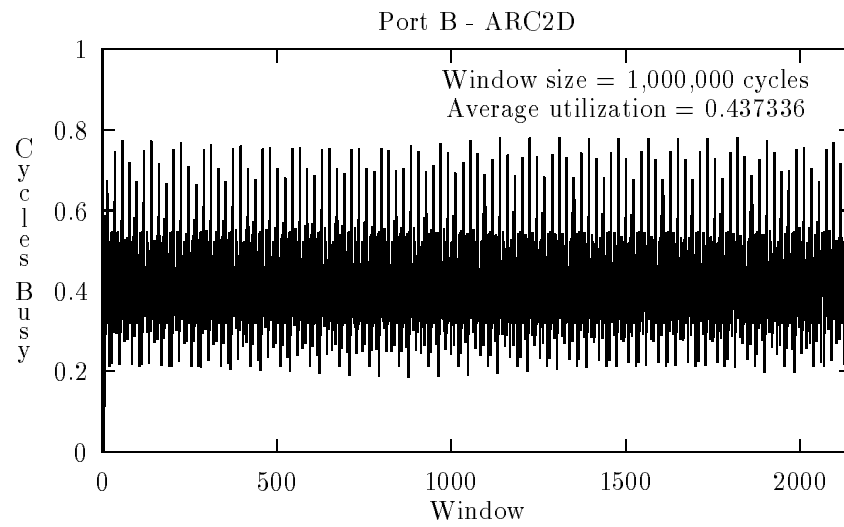
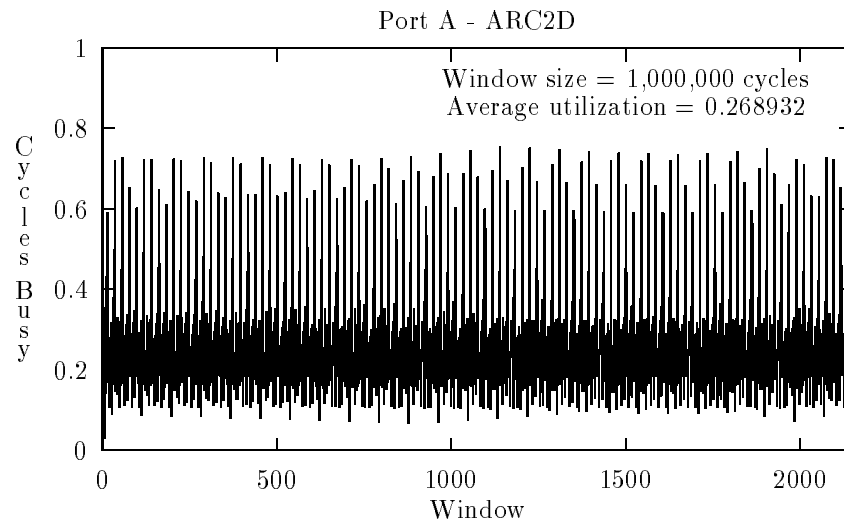


Figure 4: Port utilization for Perfect Benchmark ARC2D.

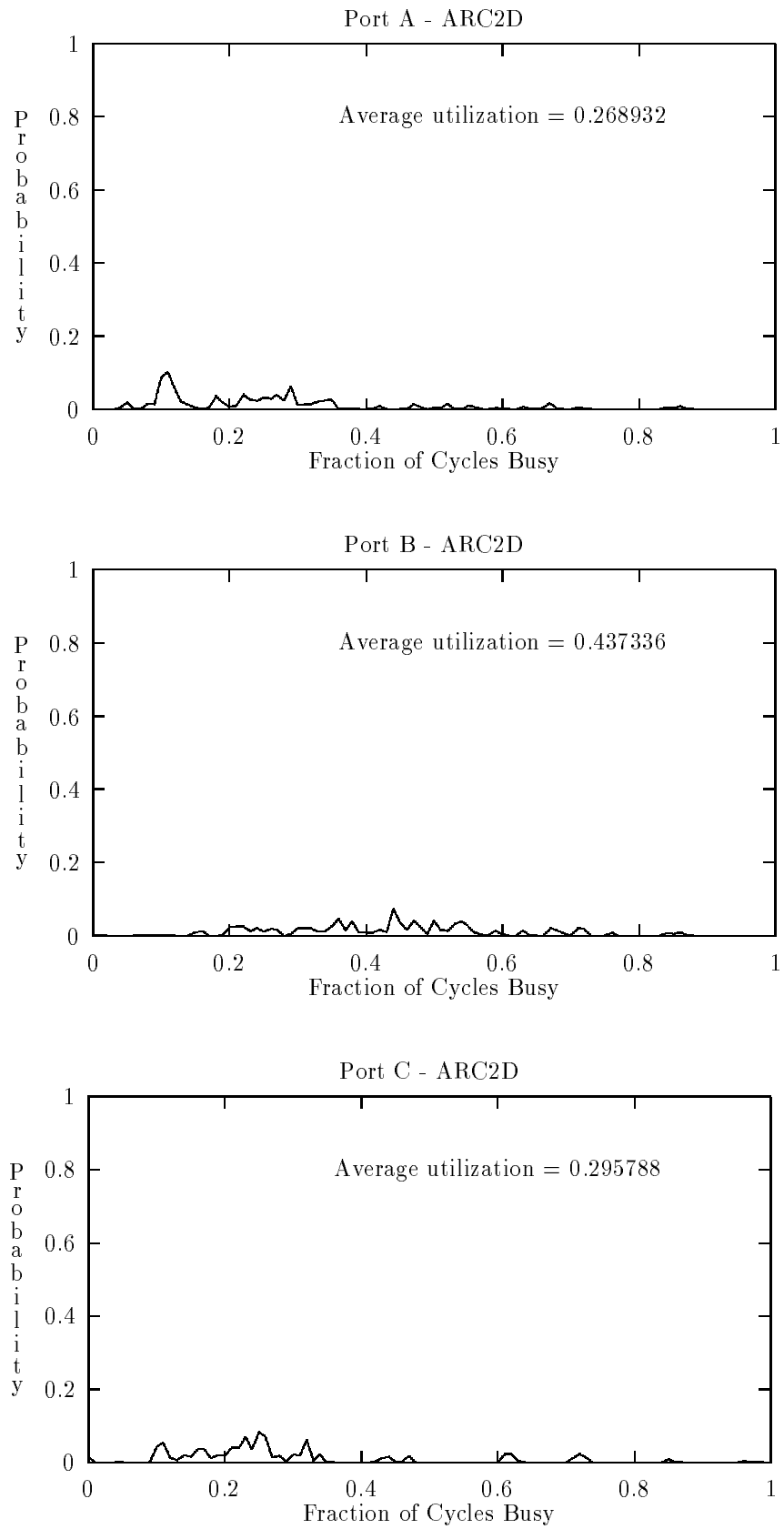


Figure 5: Port utilization histogram for Perfect Benchmark ARC2D.

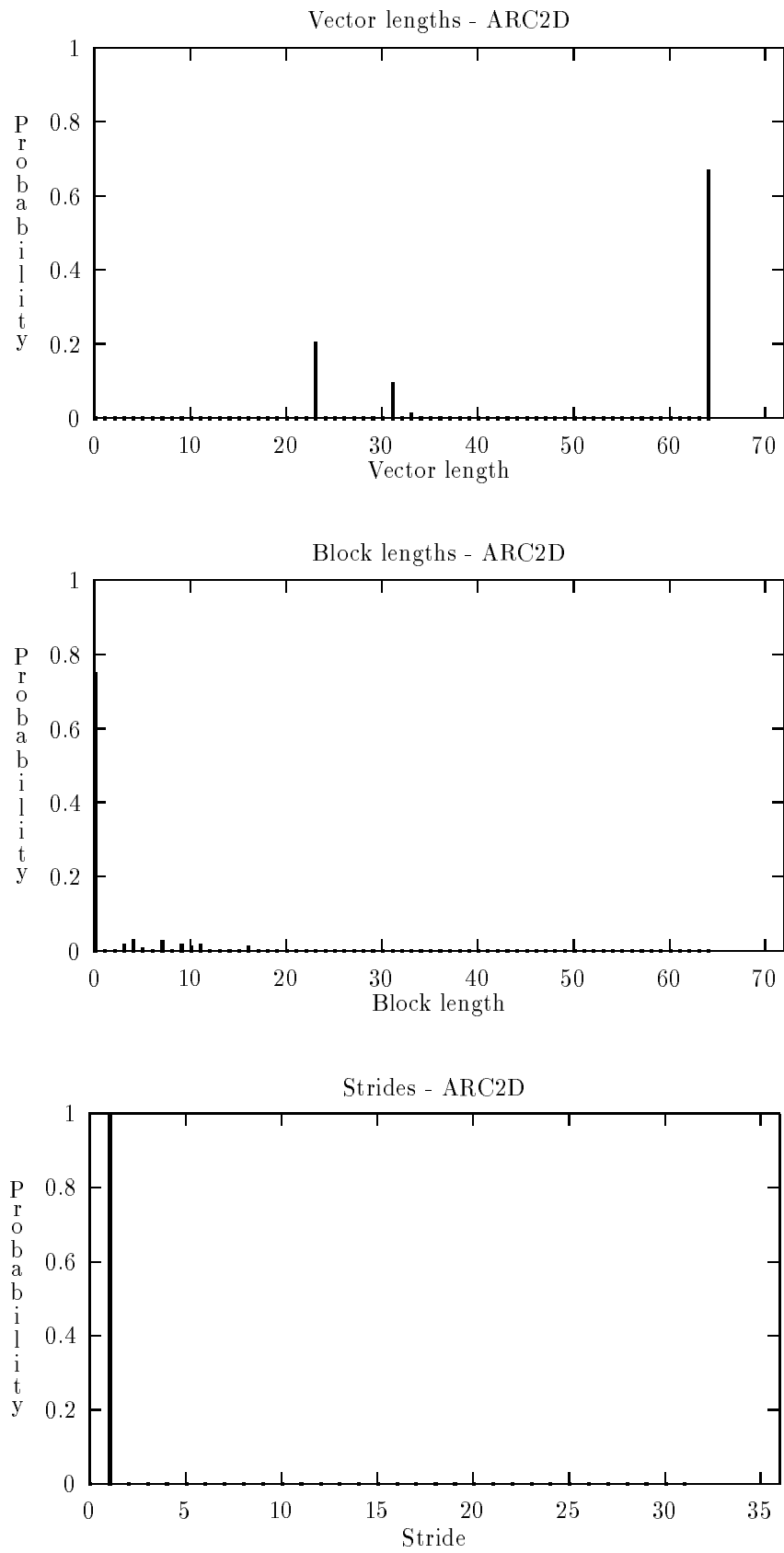


Figure 6: Distribution of lengths and strides for Perfect Benchmark ARC2D.

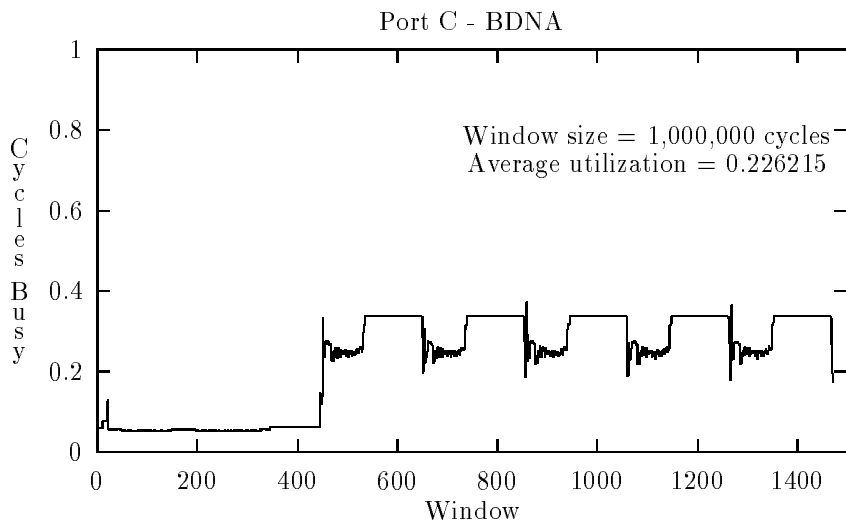
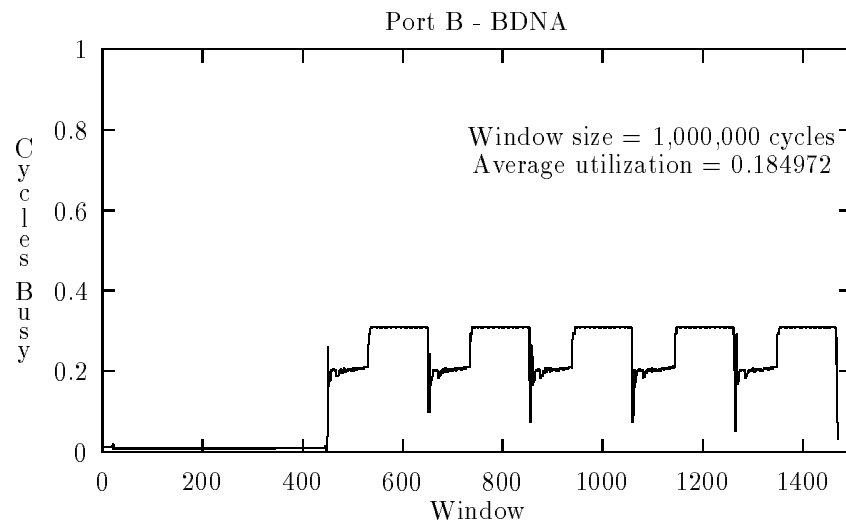
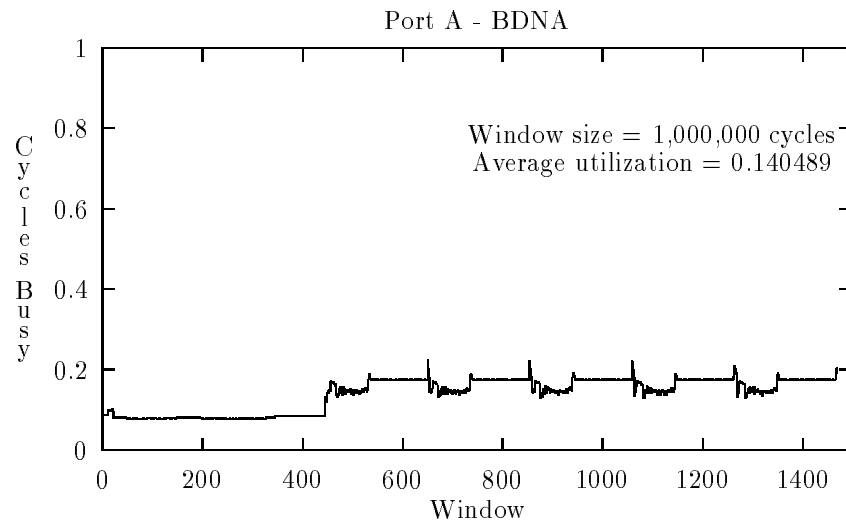


Figure 7: Port utilization for Perfect Benchmark BDNA.

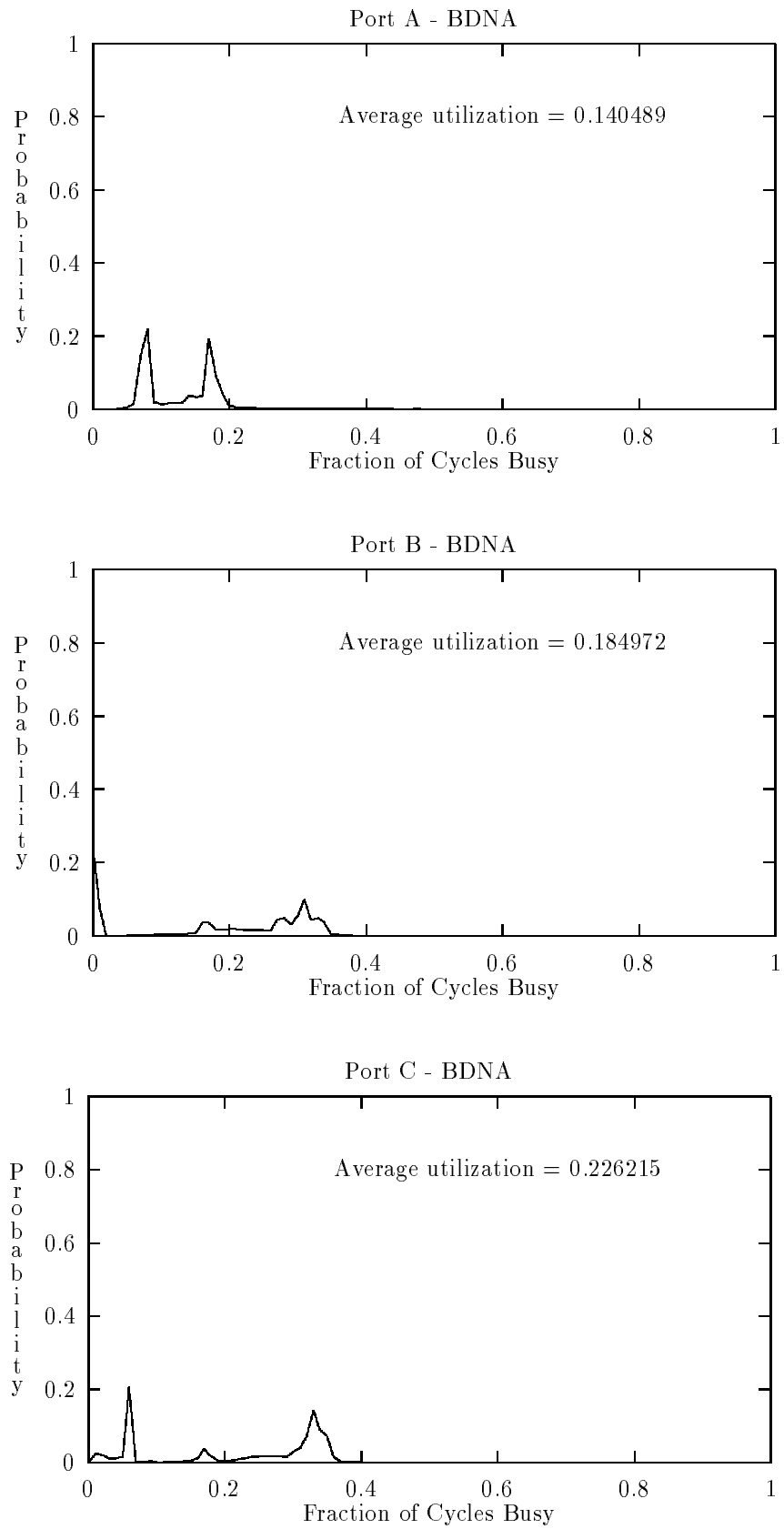


Figure 8: Port utilization histogram for Perfect Benchmark BDNA.

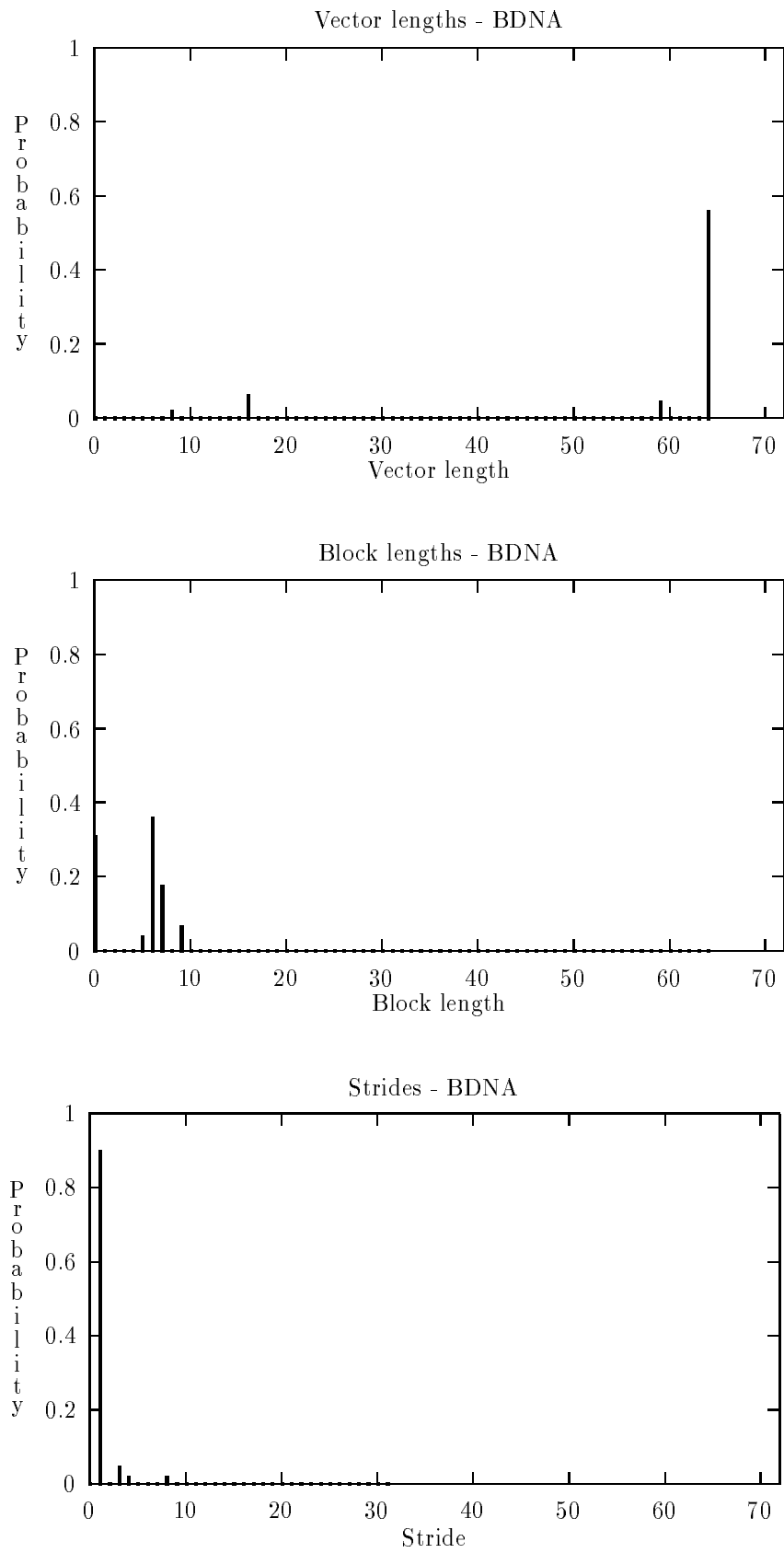


Figure 9: Distribution of lengths and strides for Perfect Benchmark BDNA

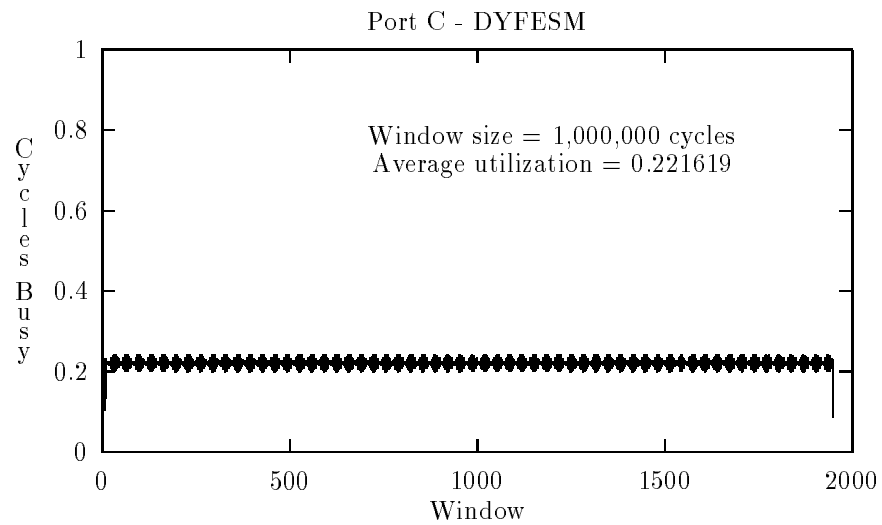
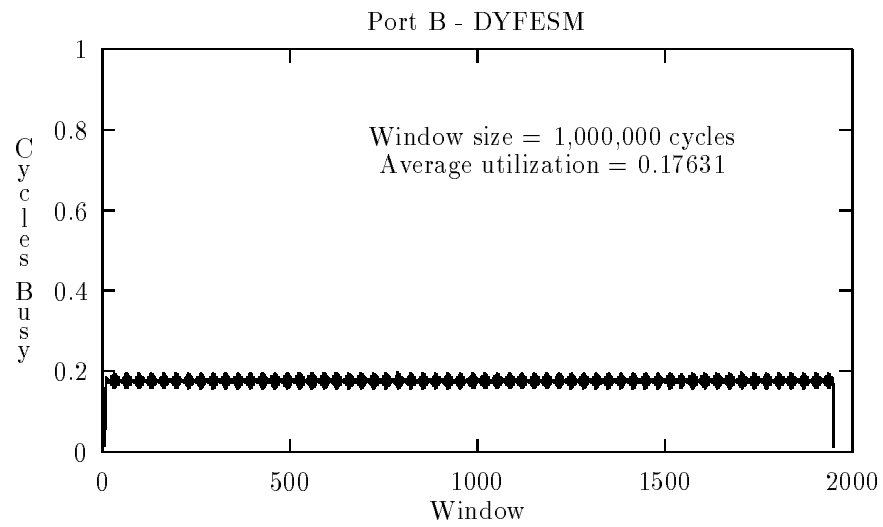
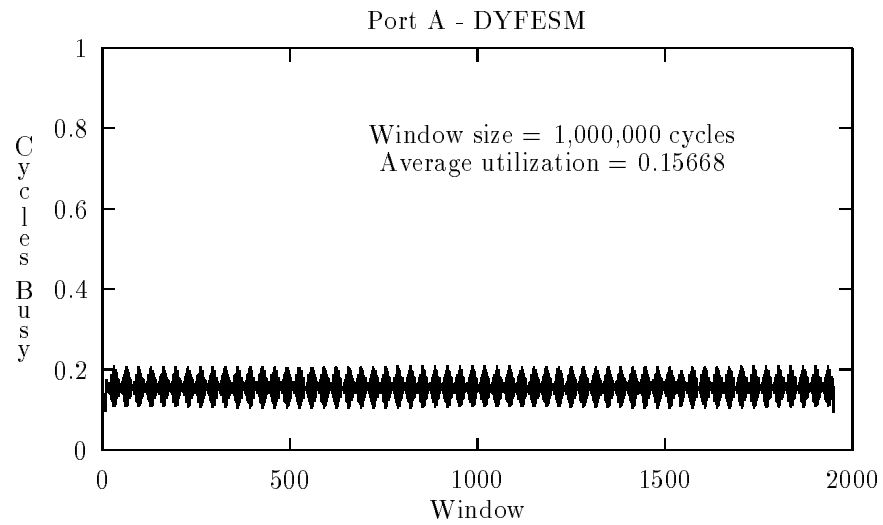


Figure 10: Port utilization for Perfect Benchmark DYFESM

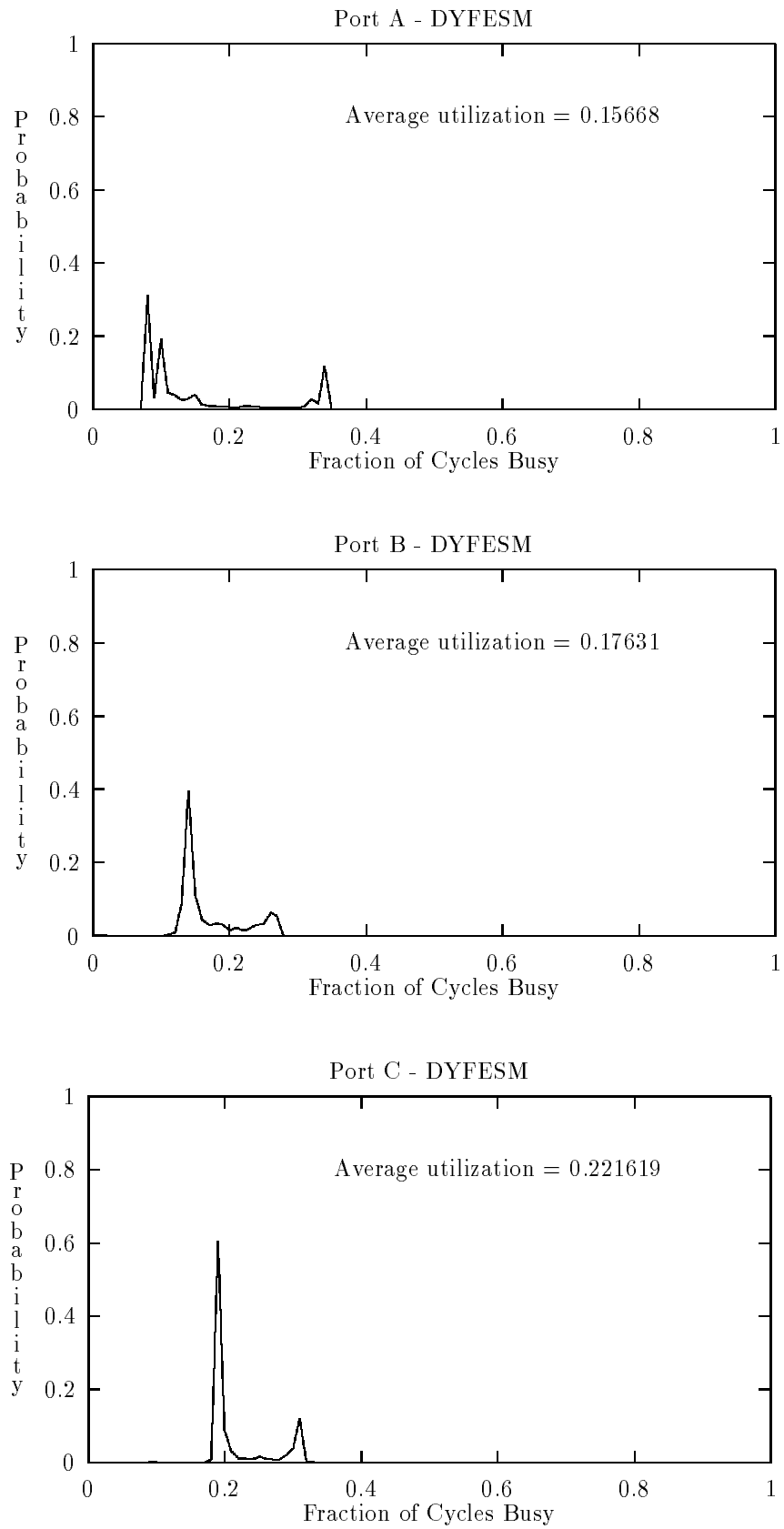


Figure 11: Port utilization histogram for Perfect Benchmark DYFESM

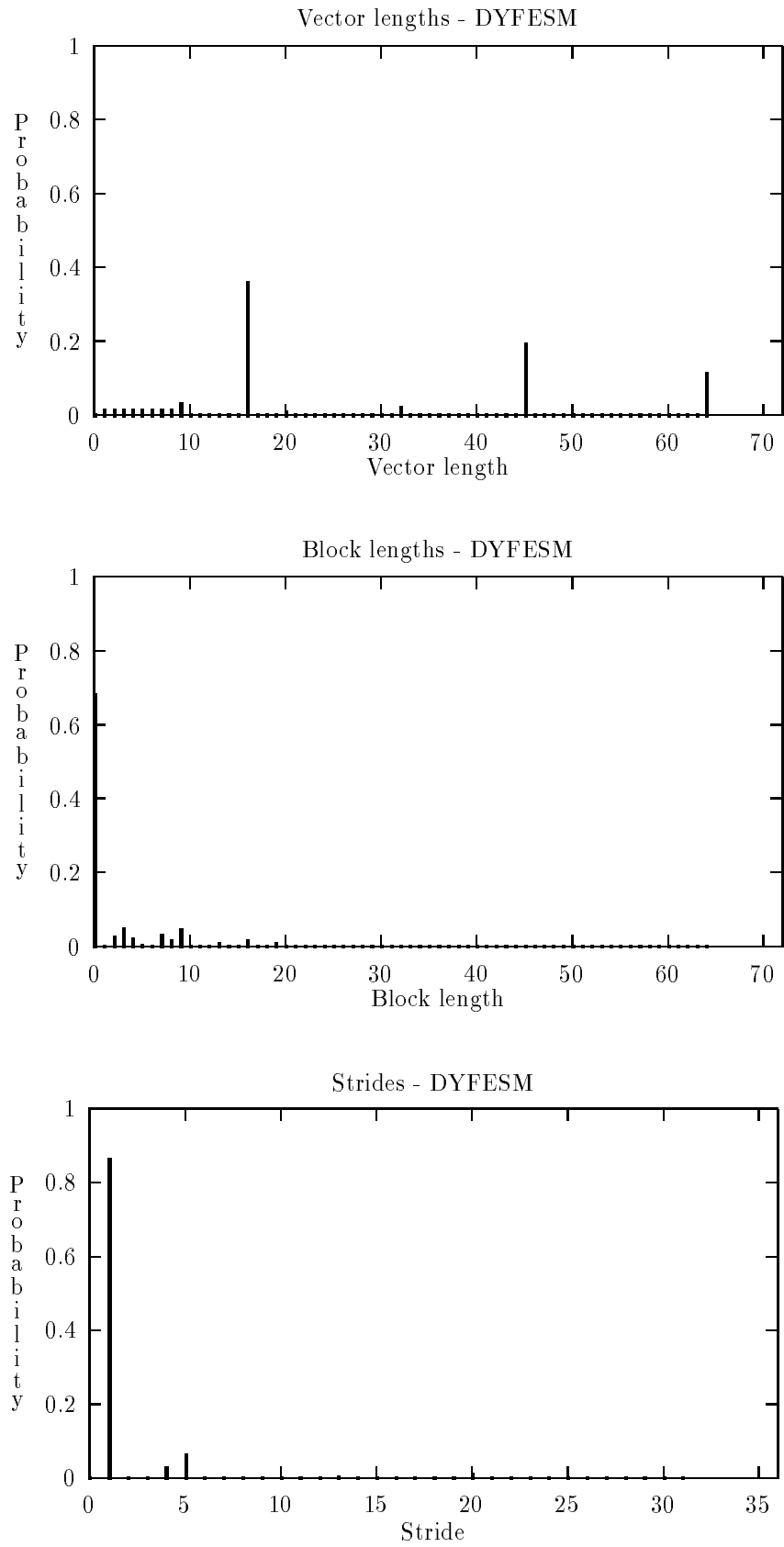


Figure 12: Distribution of lengths and strides for Perfect Benchmark DYFESM.

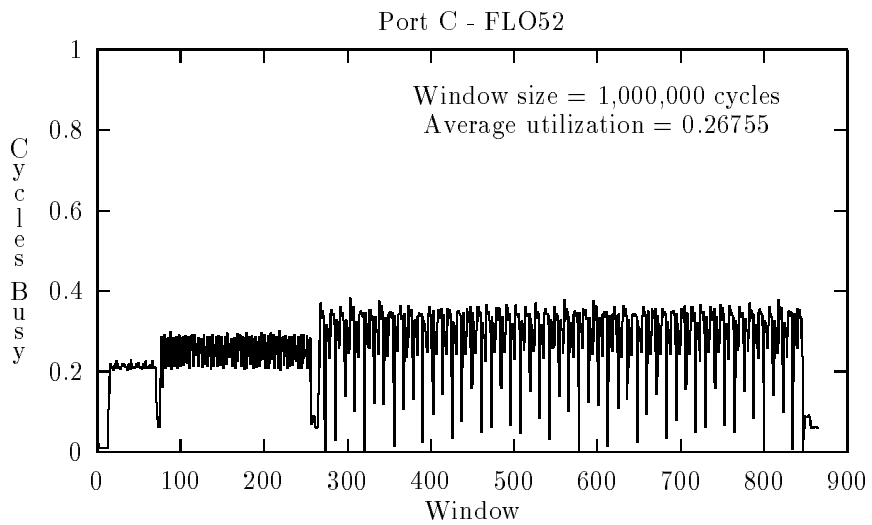
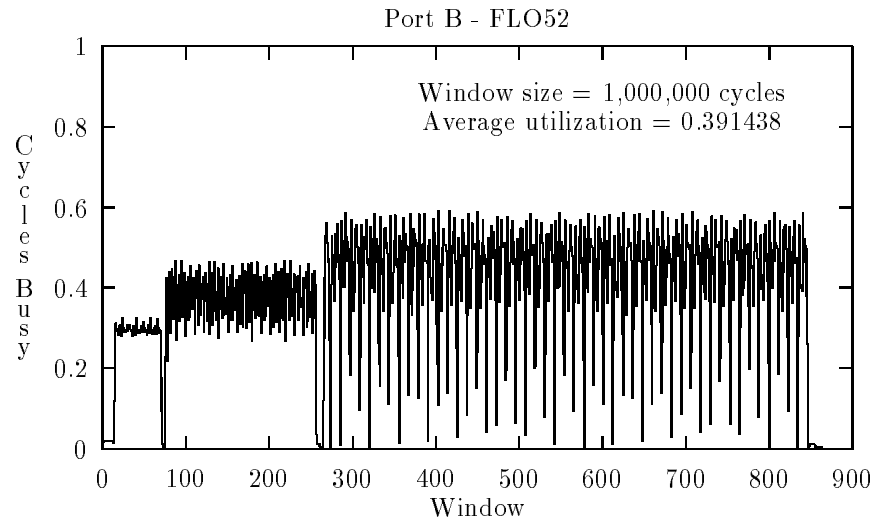
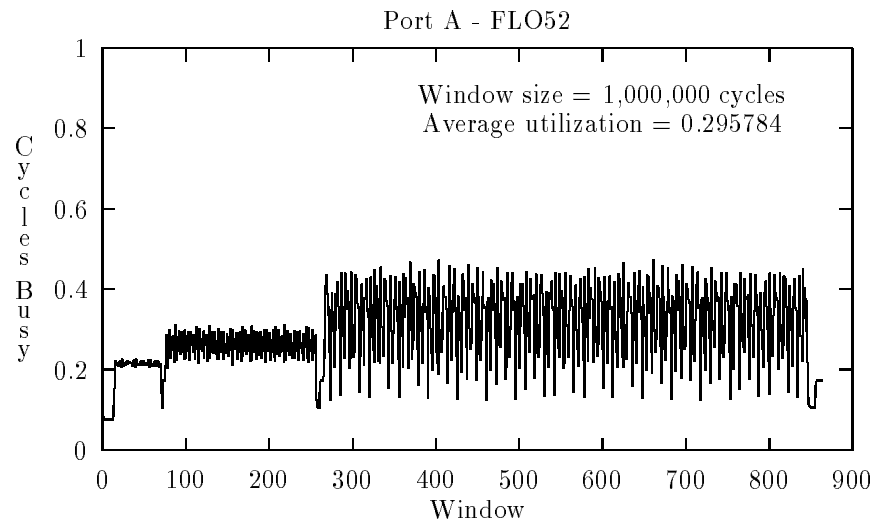


Figure 13: Port utilization for Perfect Benchmark FLO52.

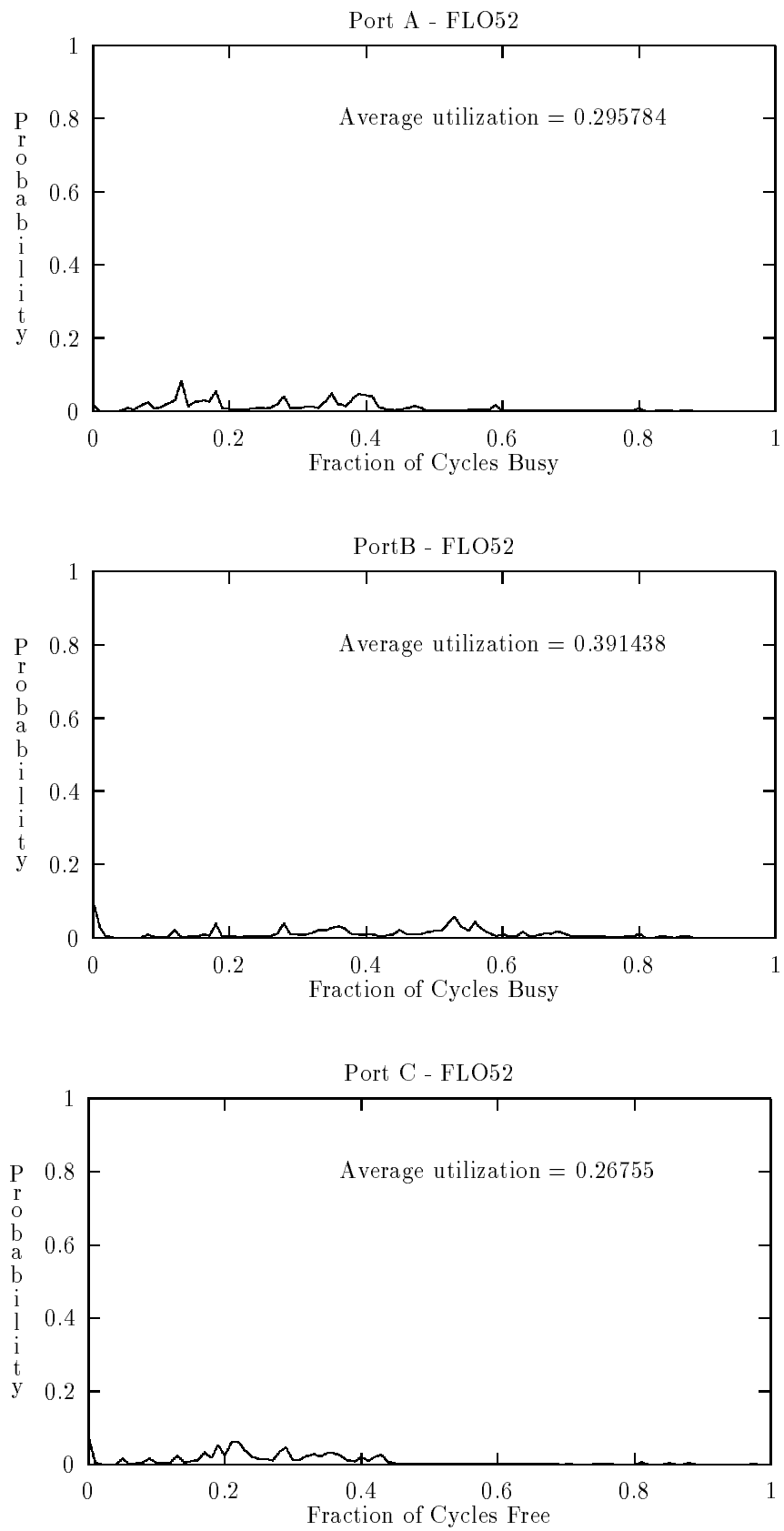


Figure 14: Port utilization histogram for Perfect Benchmark FLO52.

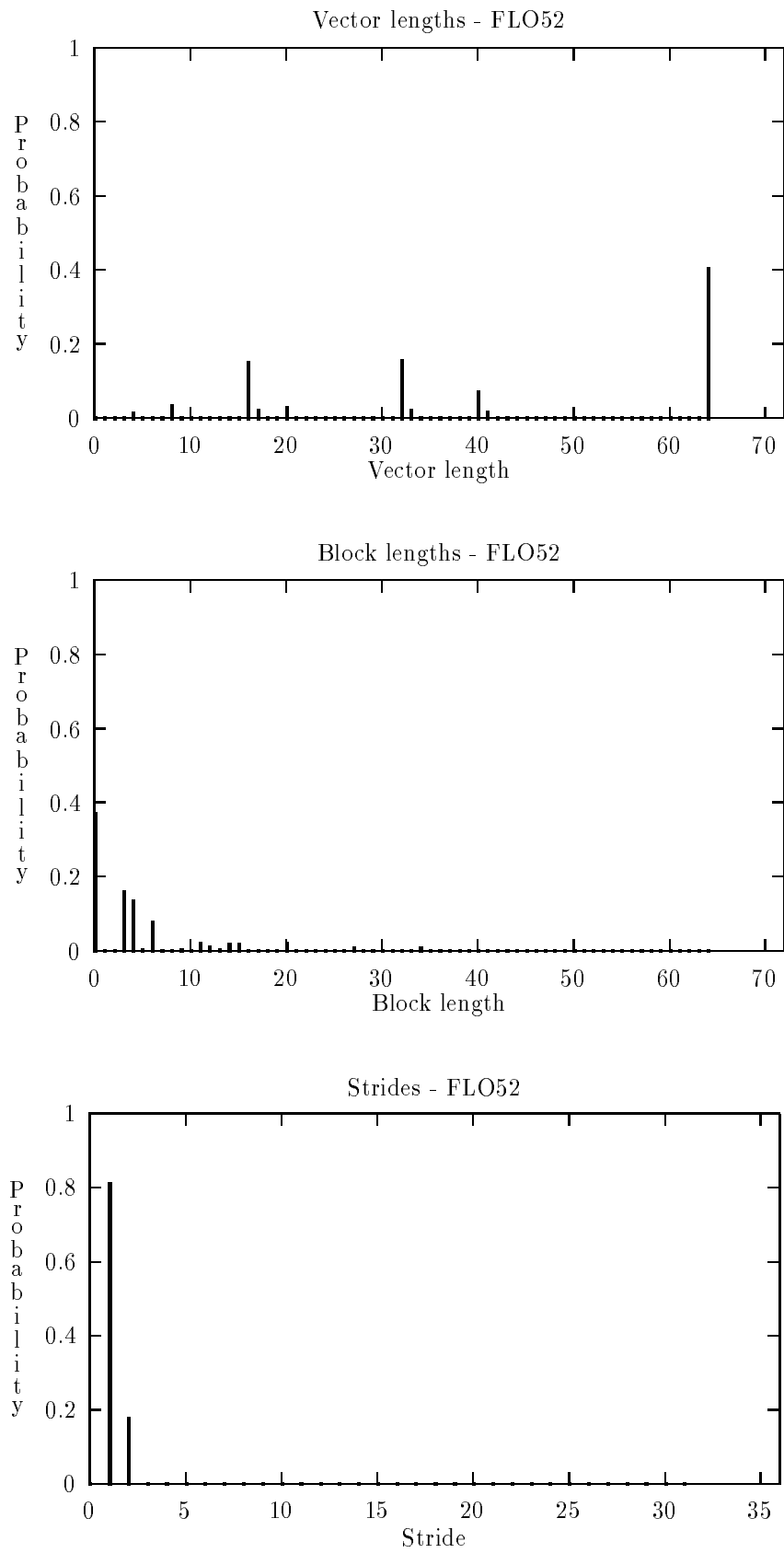


Figure 15: Distribution of lengths and strides for Perfect Benchmark FLO52.

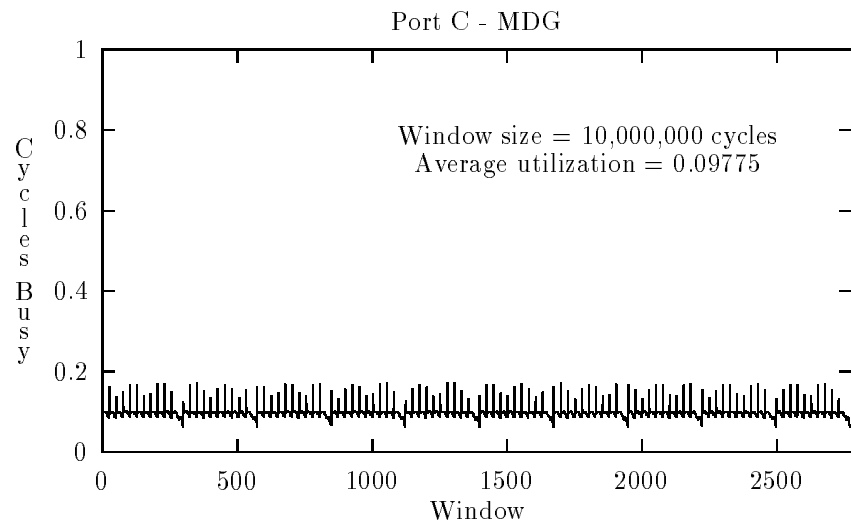
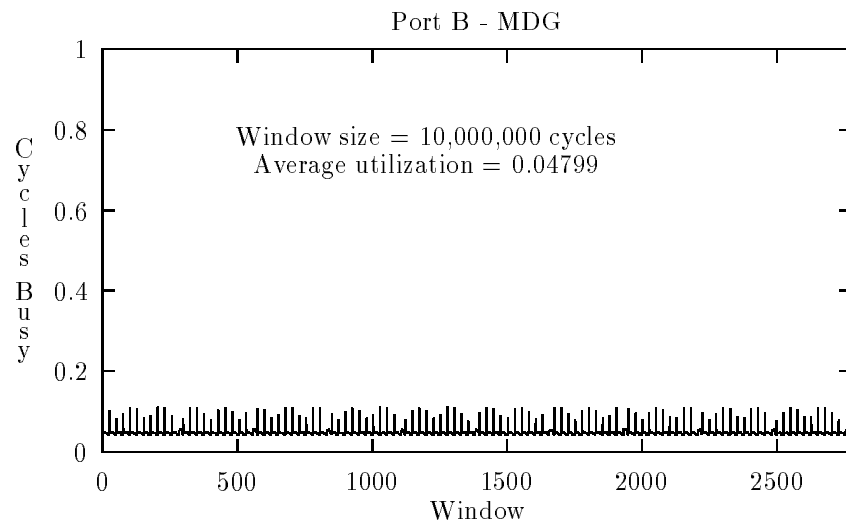
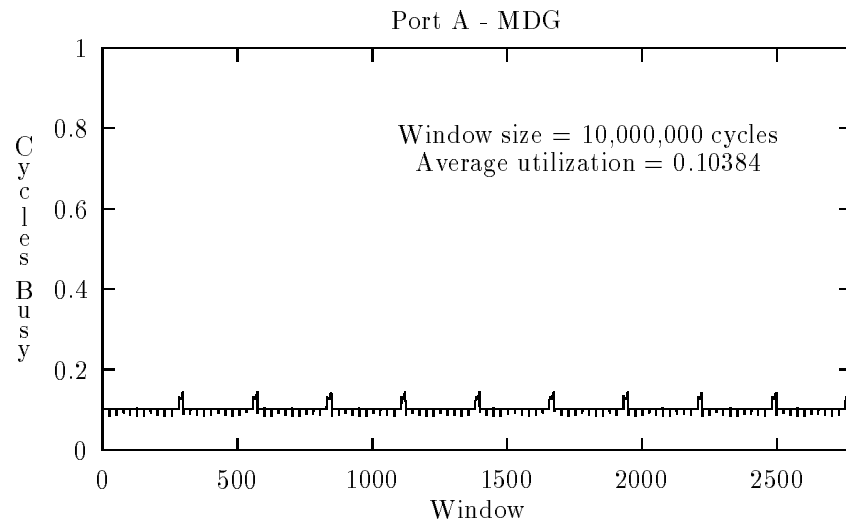


Figure 16: Port utilization for Perfect Benchmark MDG.

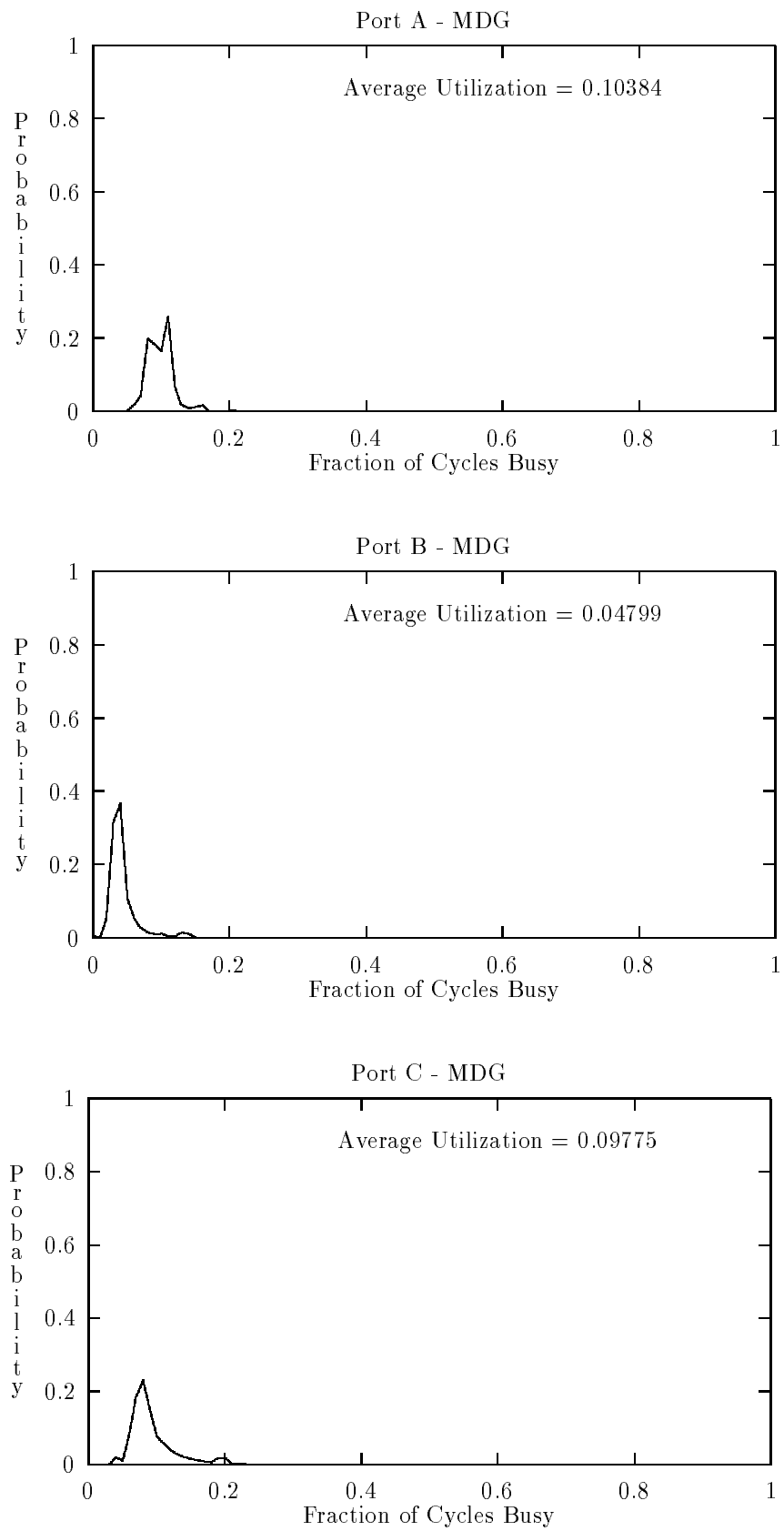


Figure 17: Port utilization histogram for Perfect Benchmark MDG.

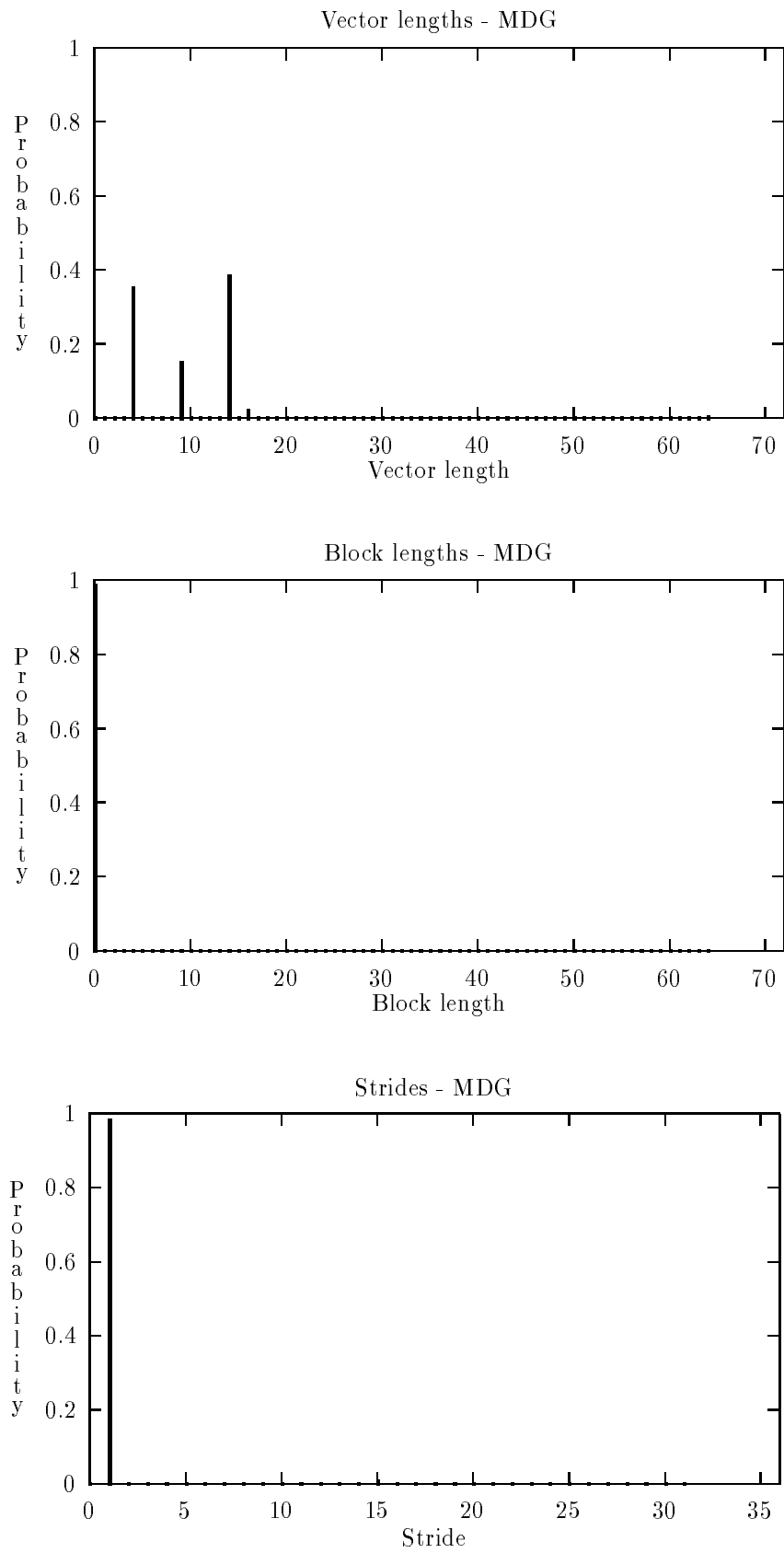


Figure 18: Distribution of lengths and strides for Perfect Benchmark MDG.

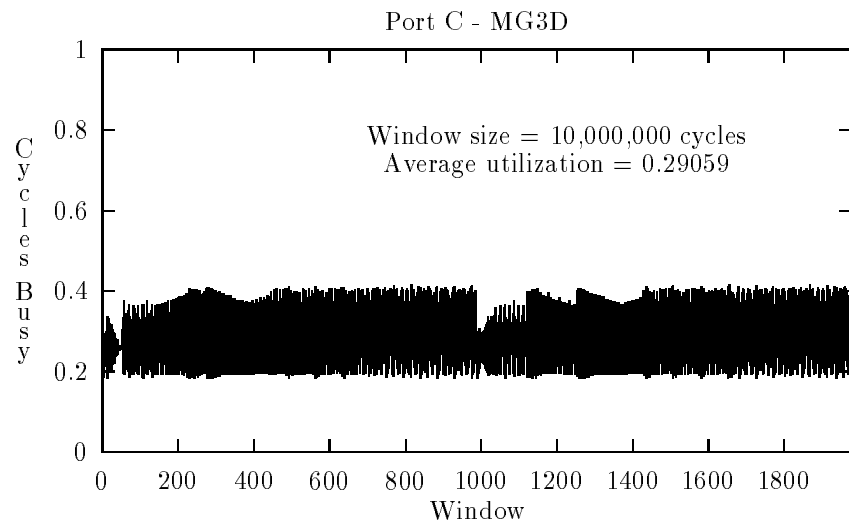
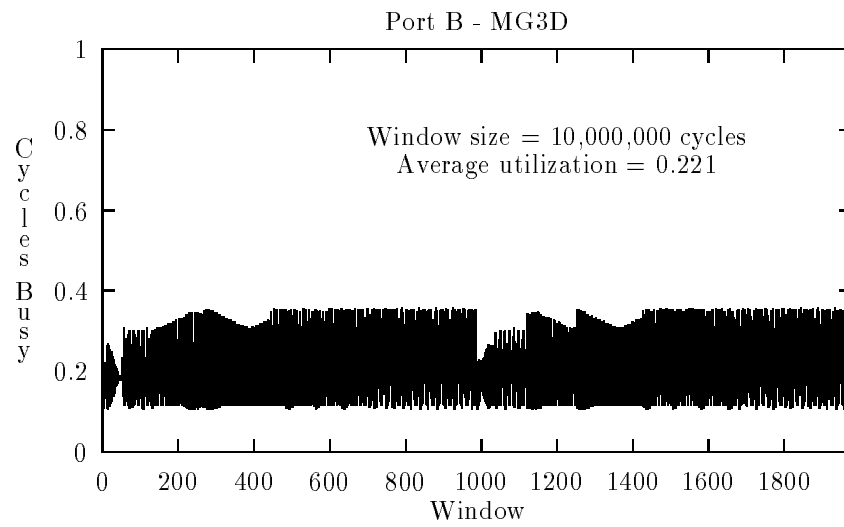
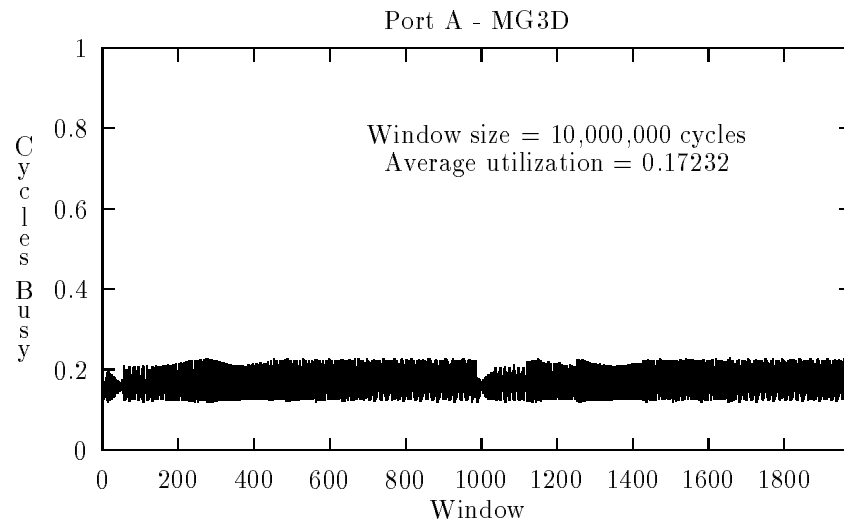


Figure 19: Port utilization for Perfect Benchmark MG3D.

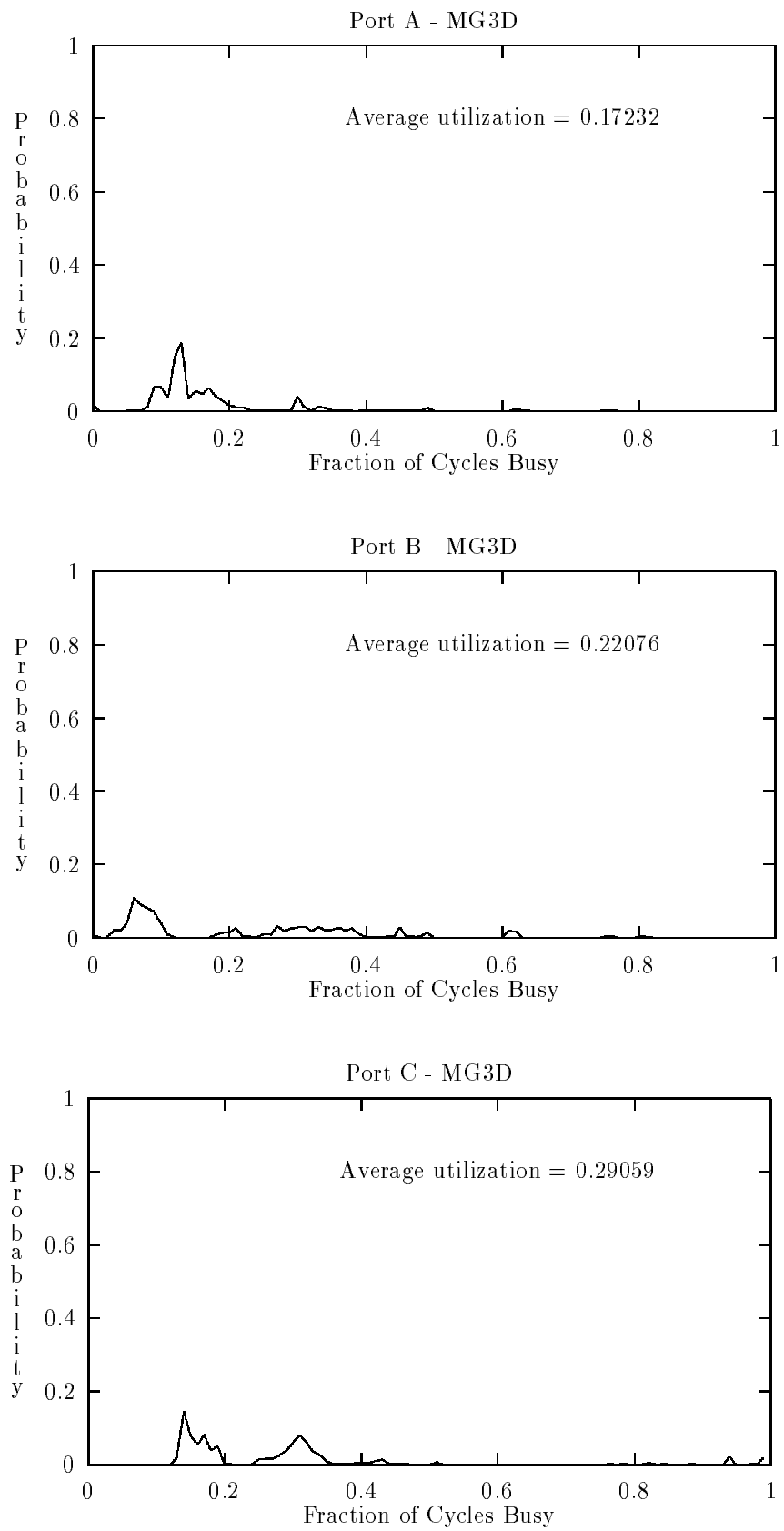


Figure 20: Port utilization histogram for Perfect Benchmark MG3D.

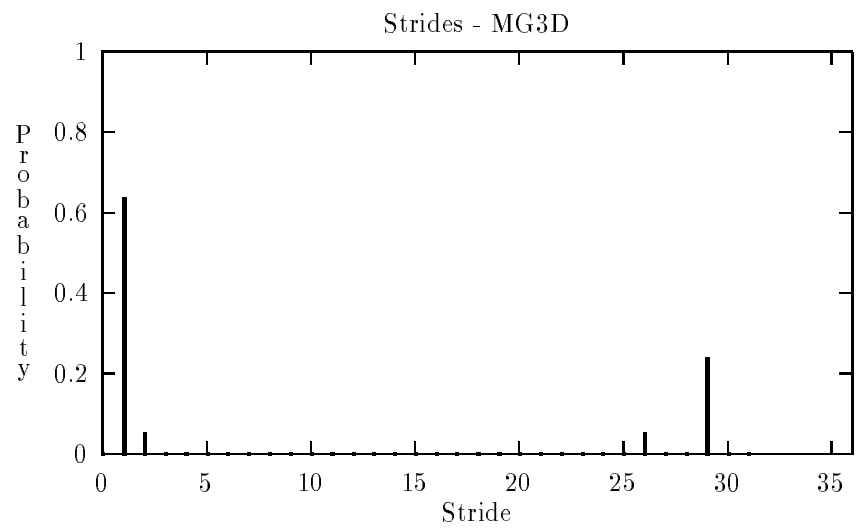
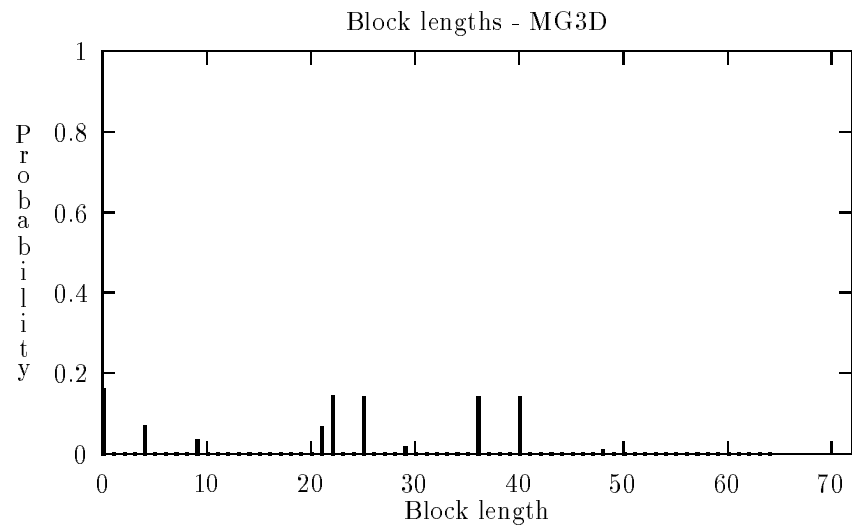
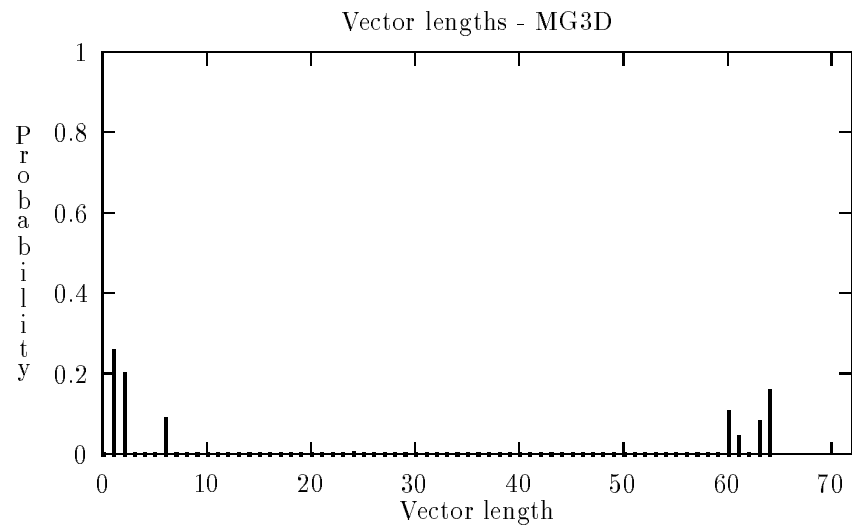


Figure 21: for Perfect Benchmark MG3D.

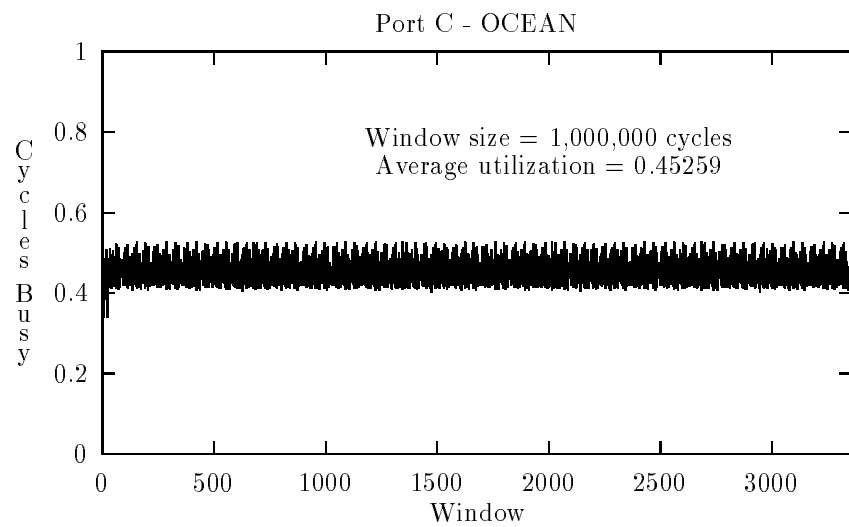
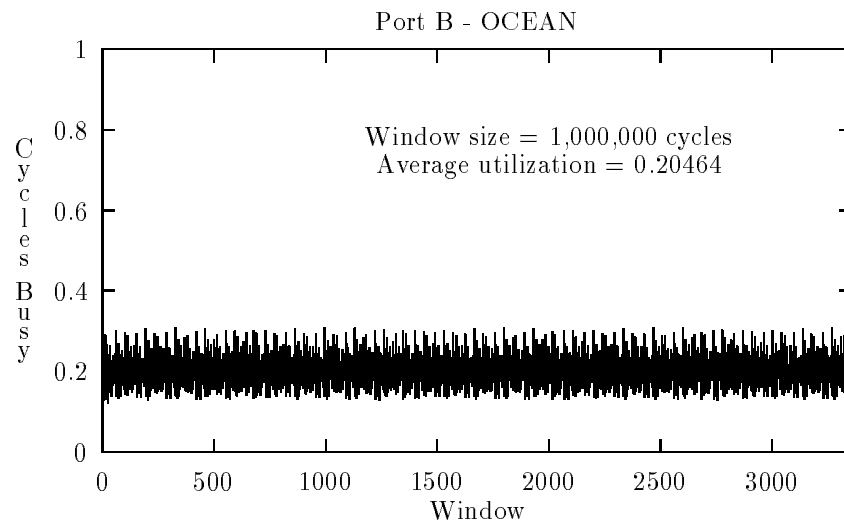
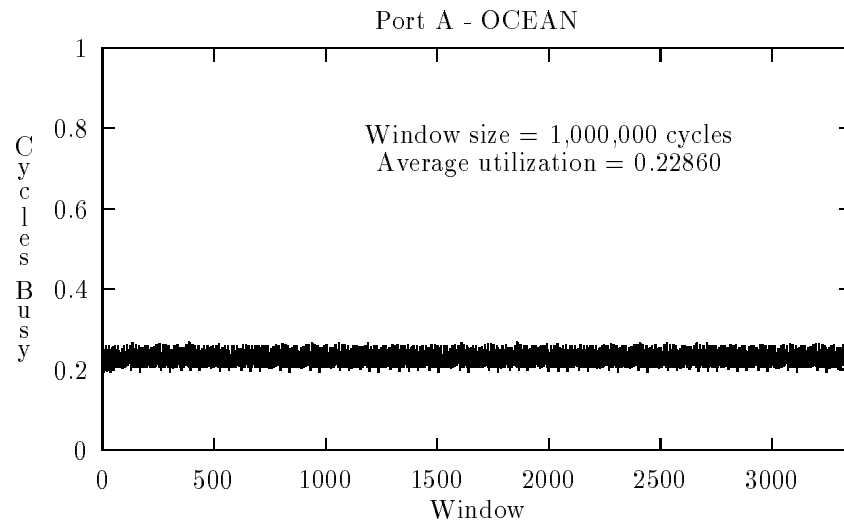


Figure 22: Port utilization for Perfect Benchmark OCEAN.

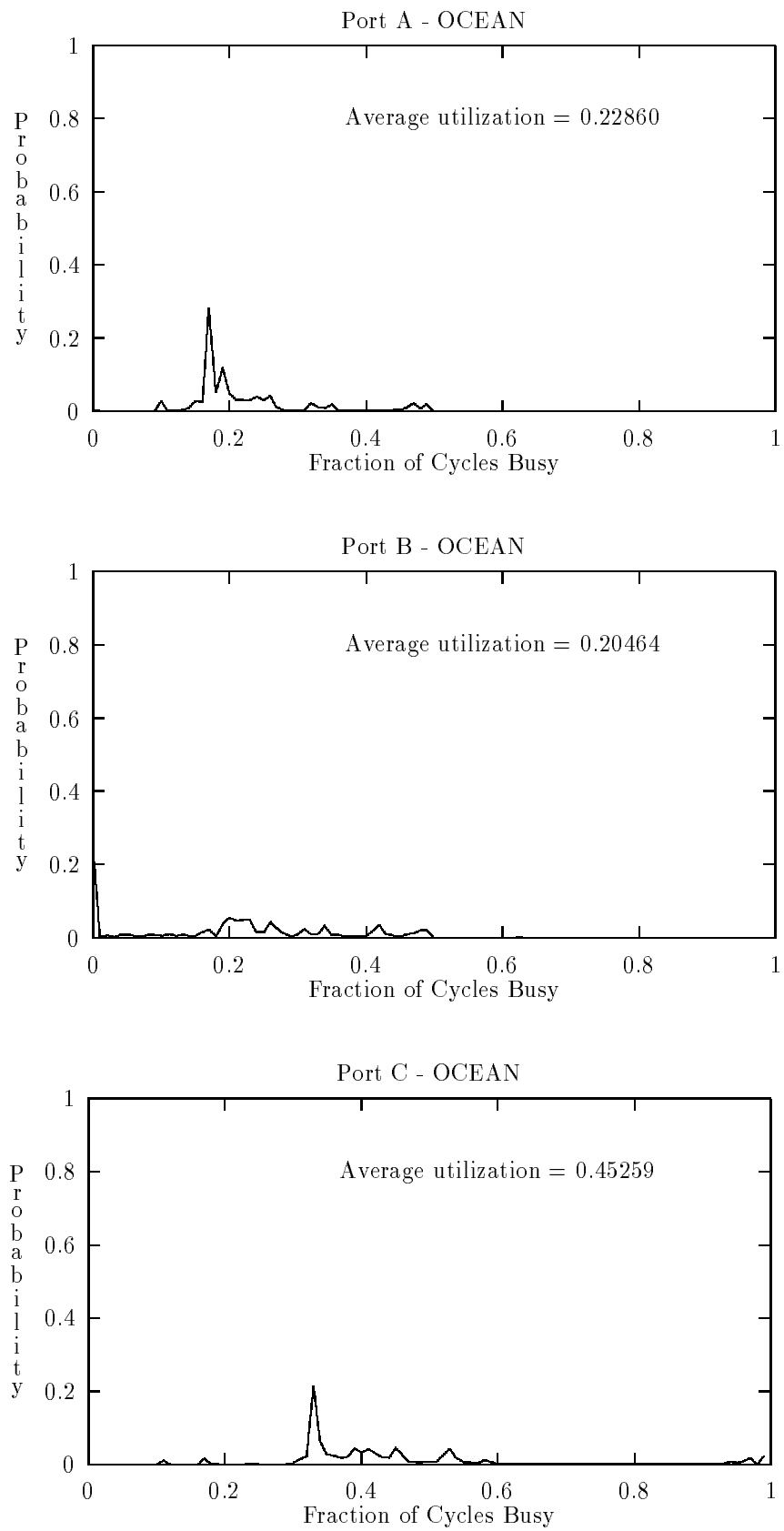


Figure 23: Port utilization histogram for Perfect Benchmark OCEAN.

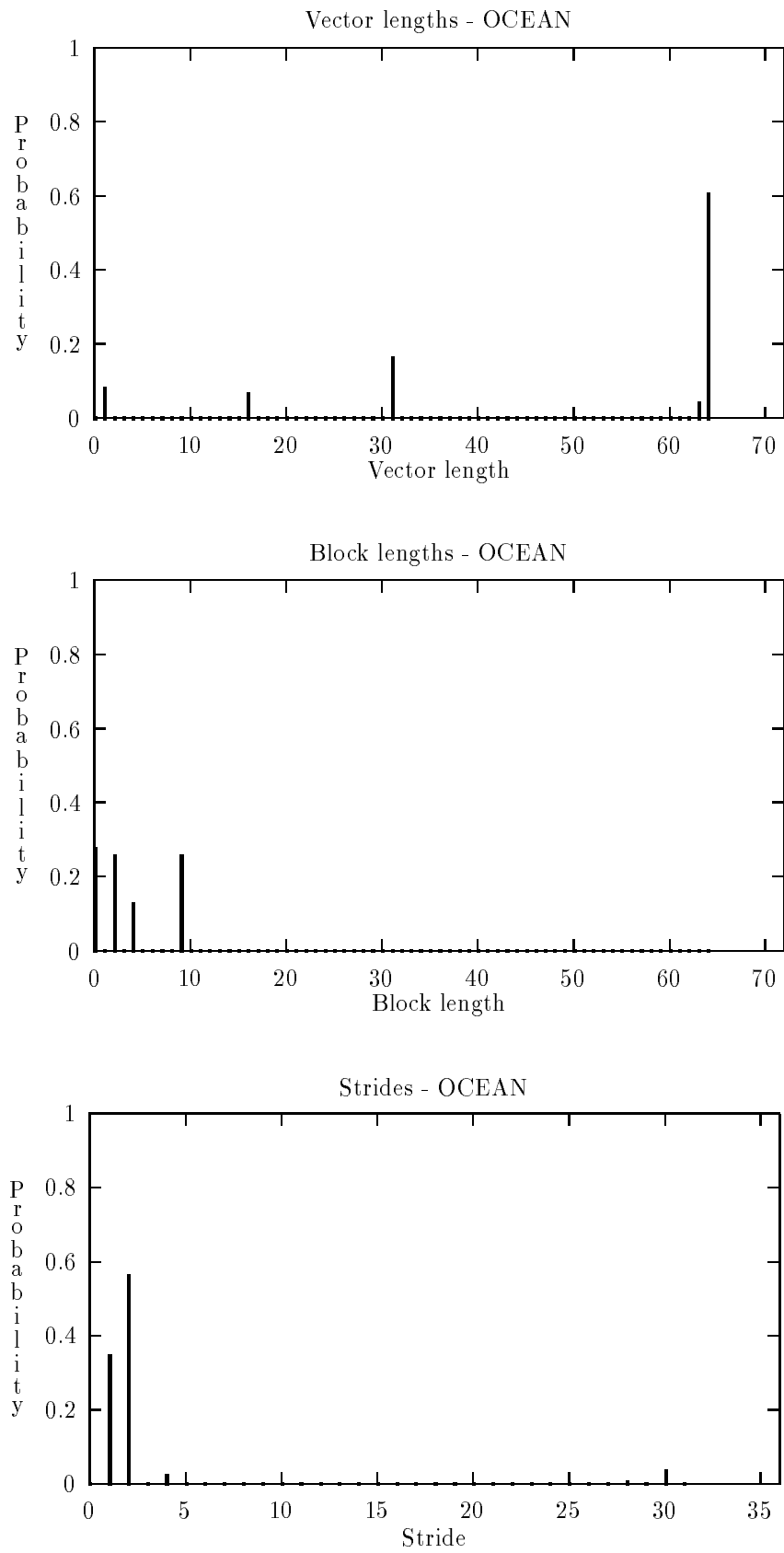


Figure 24: Distribution of lengths and strides for Perfect Benchmark OCEAN.

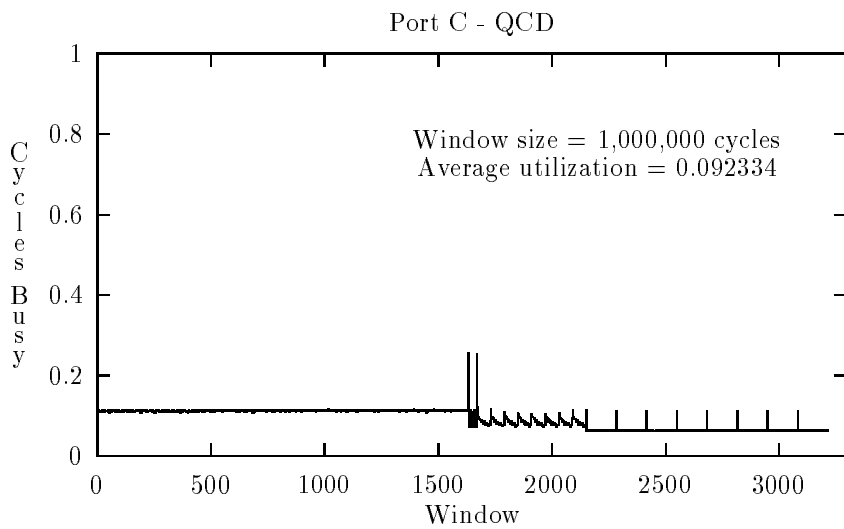
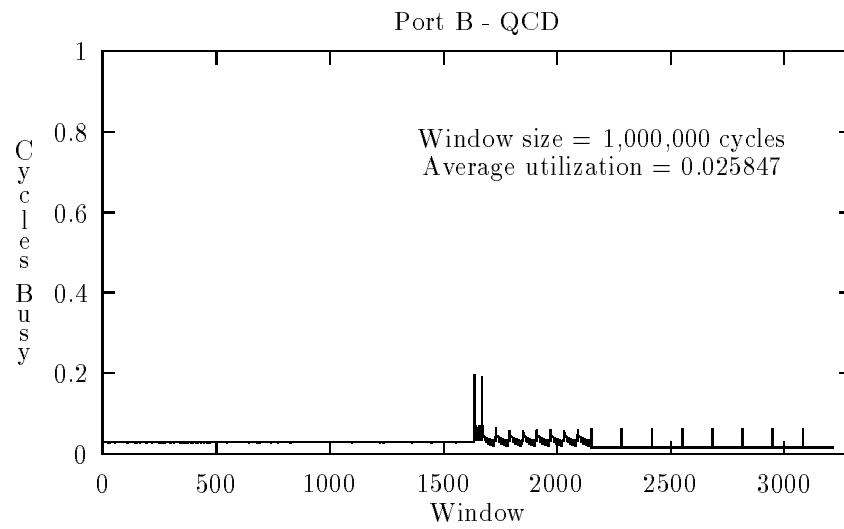
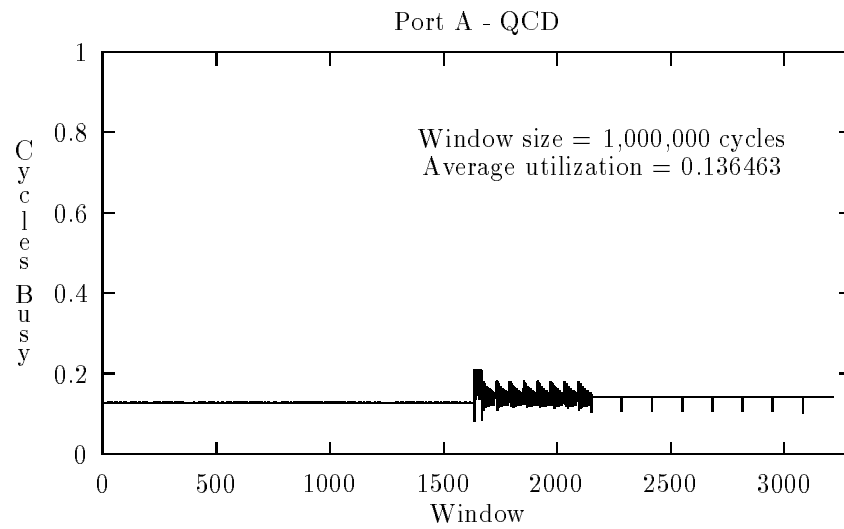


Figure 25: Port utilization for Perfect Benchmark QCD.

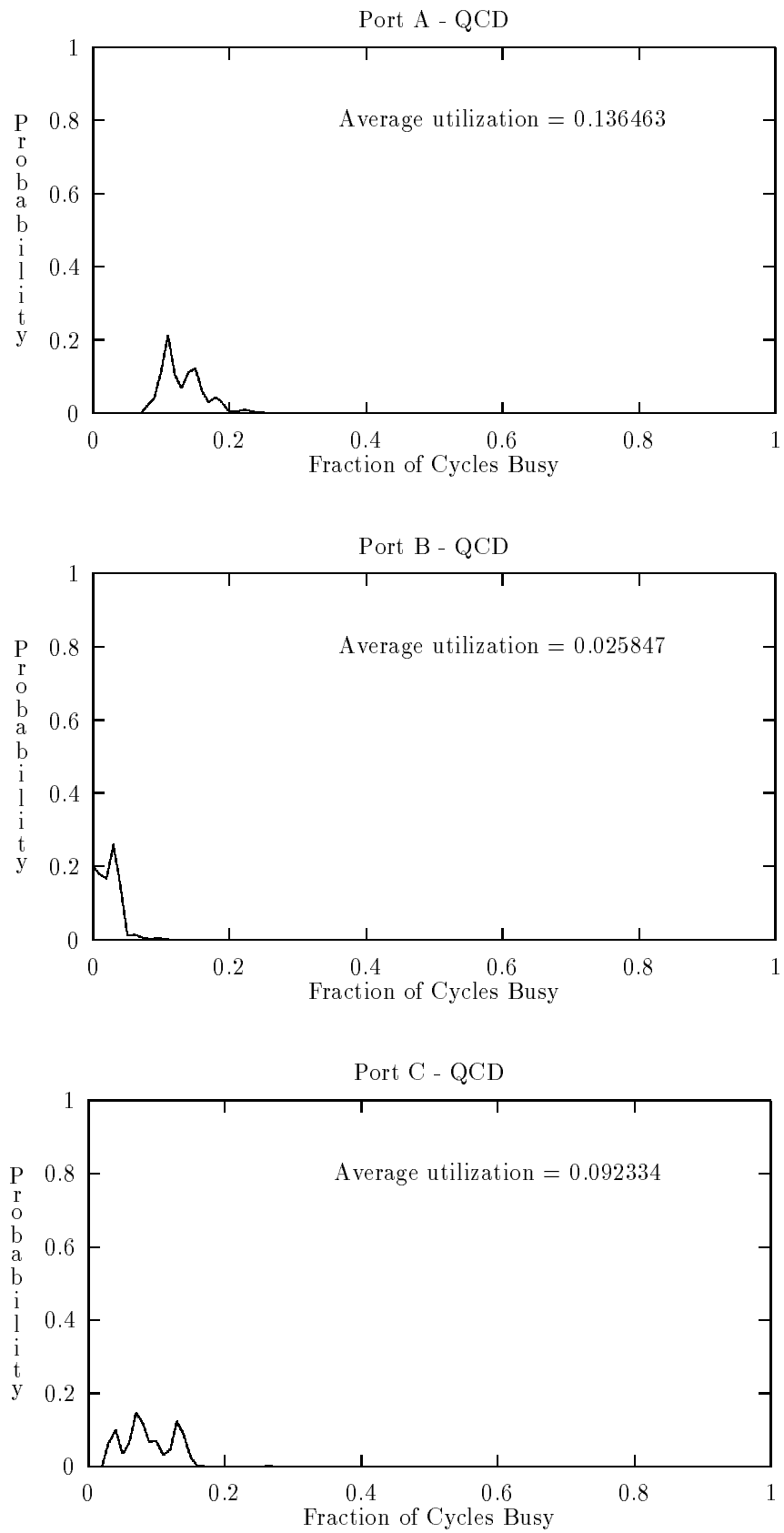


Figure 26: Port utilization histogram for Perfect Benchmark QCD.

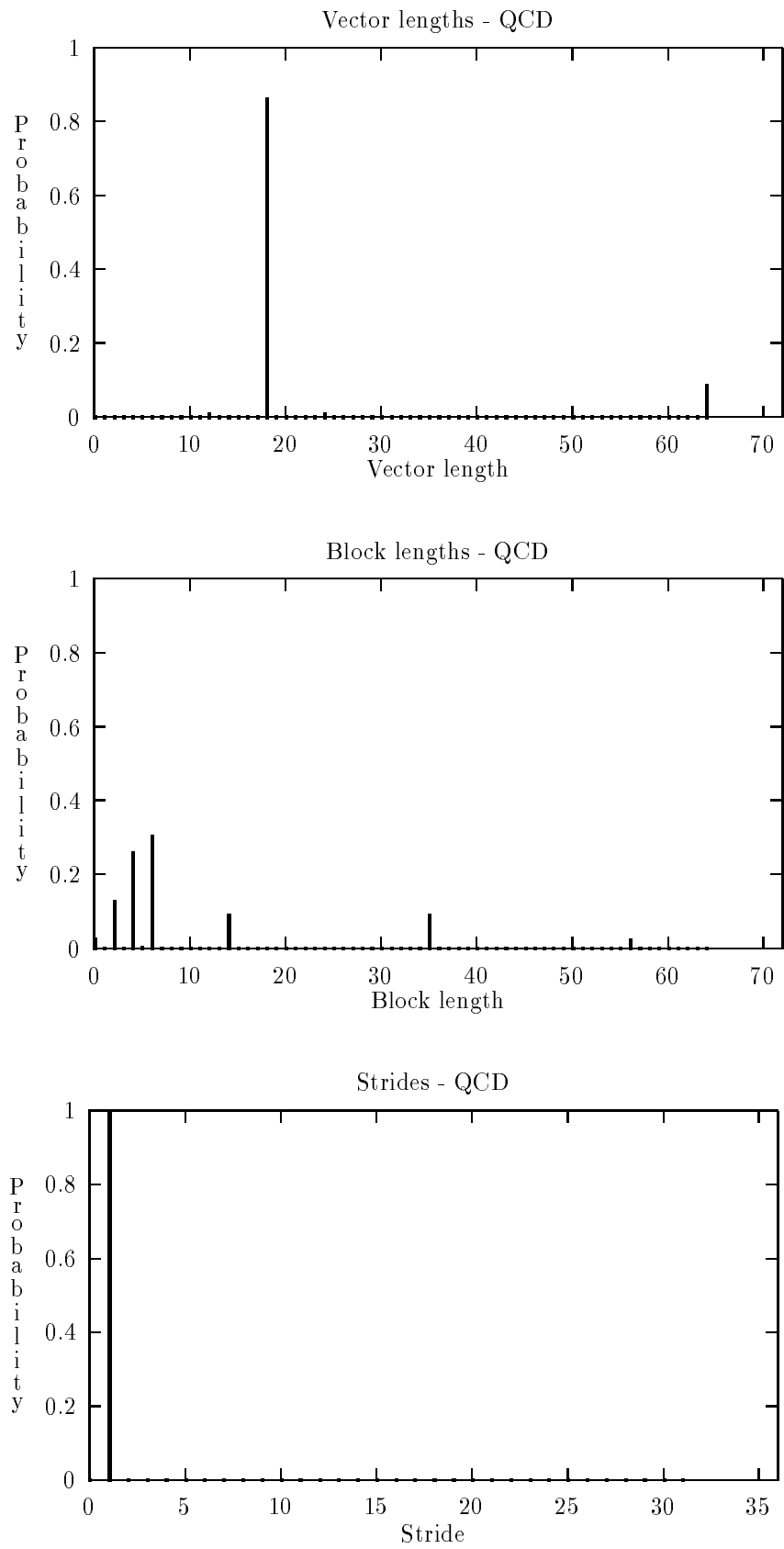


Figure 27: Distribution of lengths and strides for Perfect Benchmark QCD.

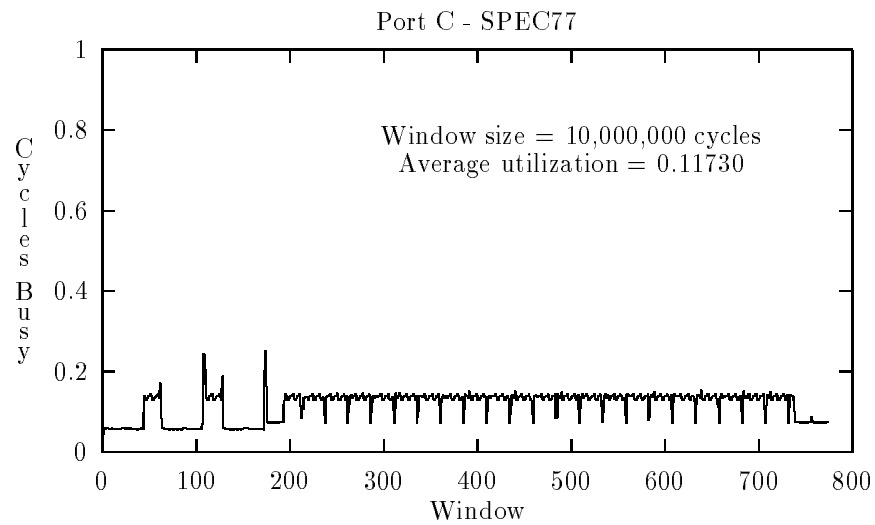
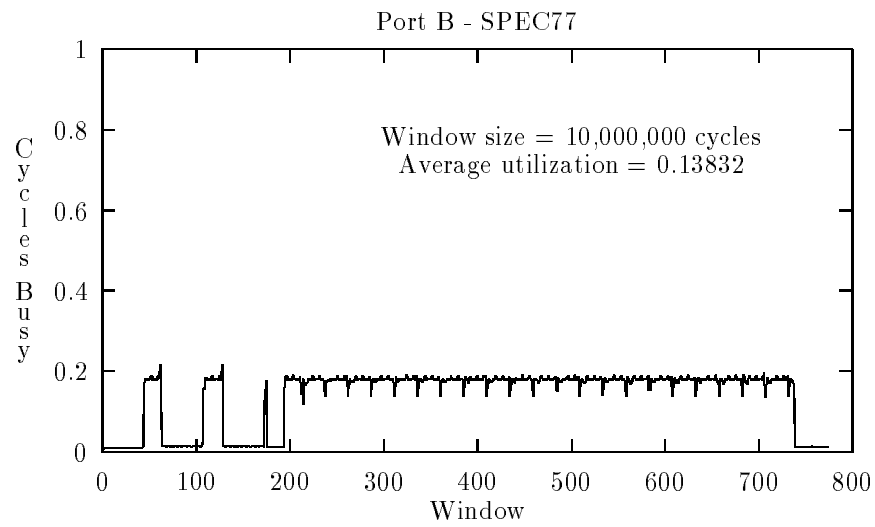
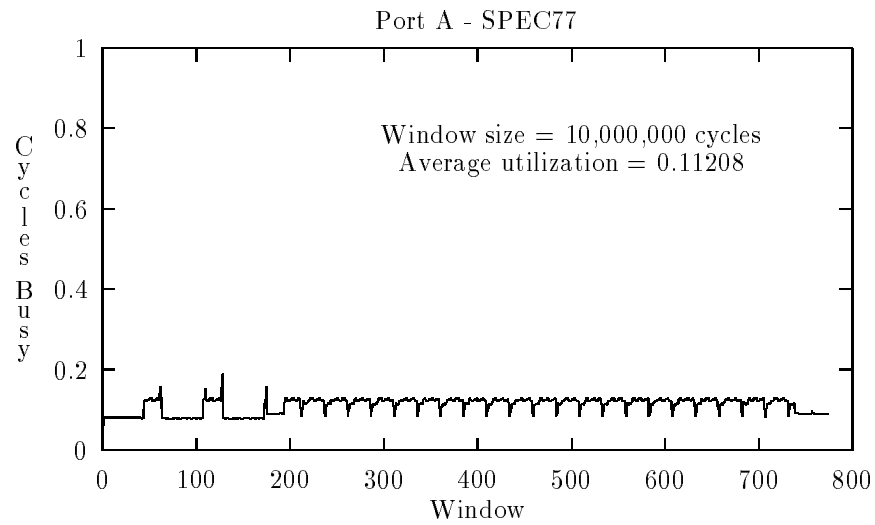


Figure 28: Port utilization for Perfect Benchmark SPEC77.

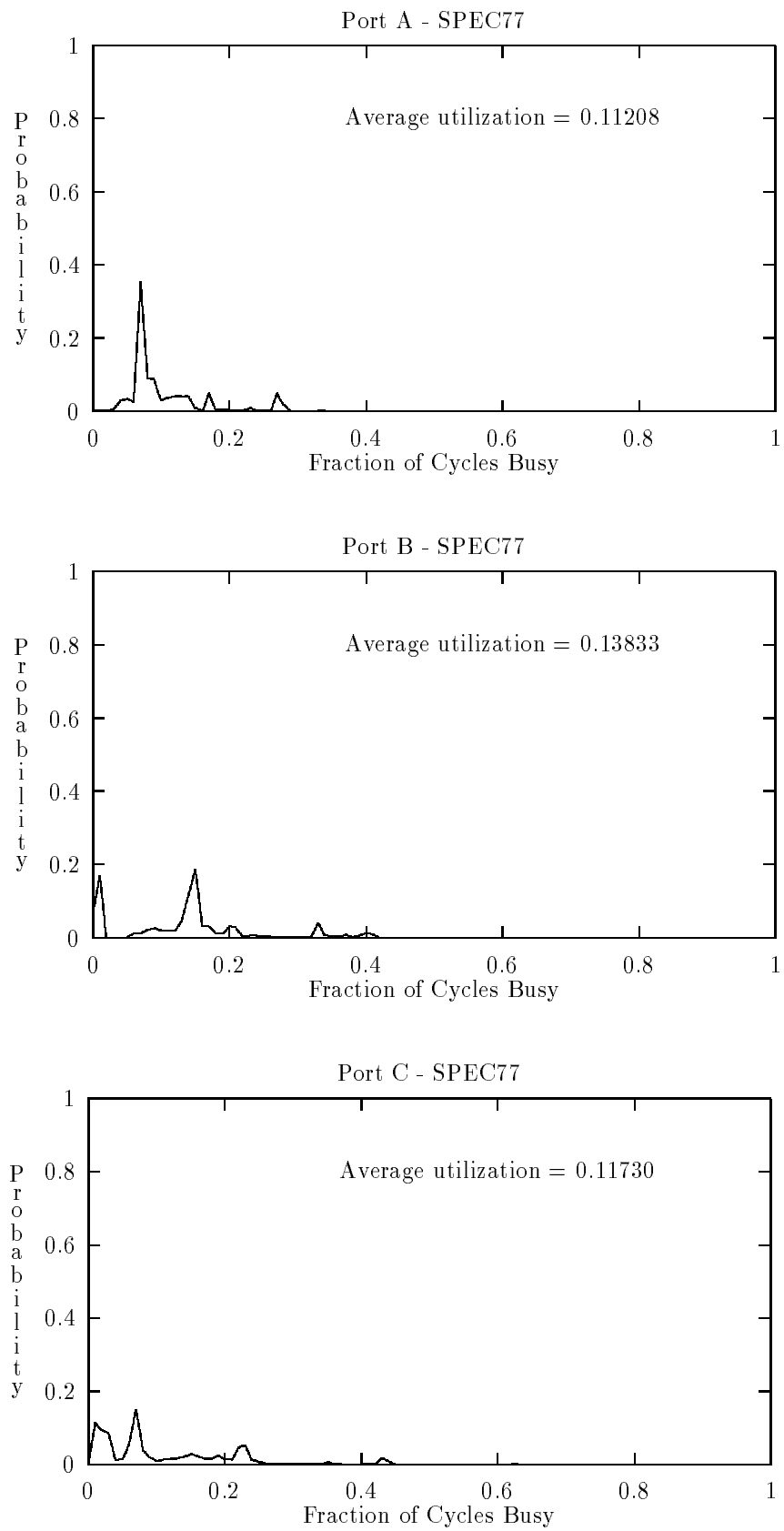


Figure 29: Port utilization histogram for Perfect Benchmark SPEC77.

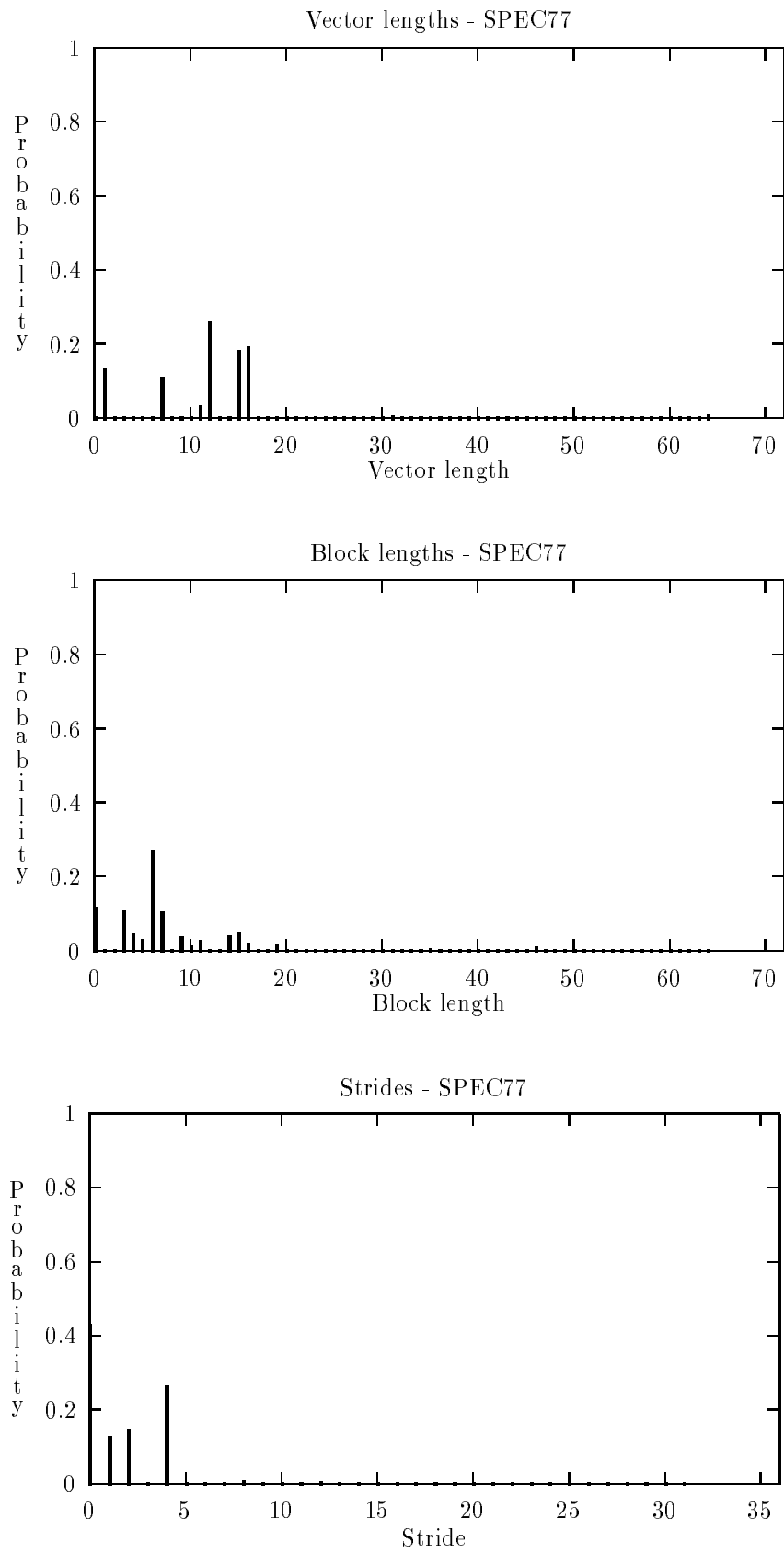


Figure 30: Distribution of lengths and strides for Perfect Benchmark SPEC77.

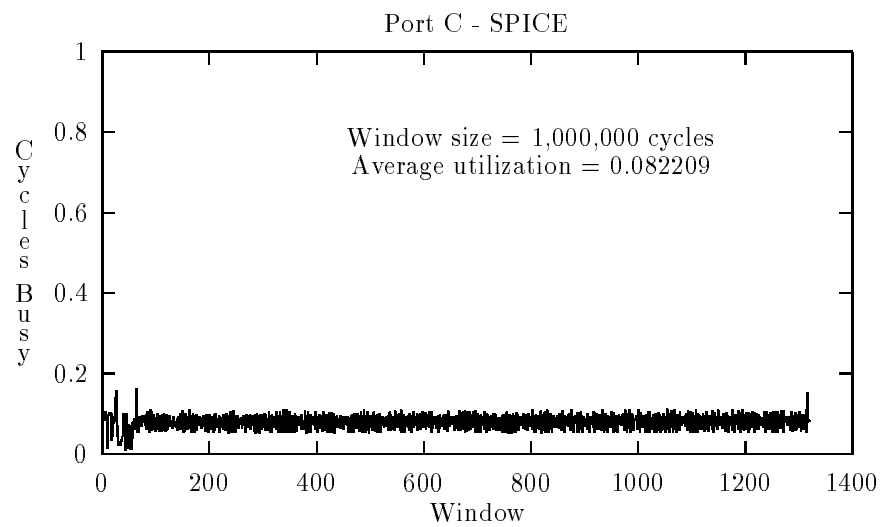
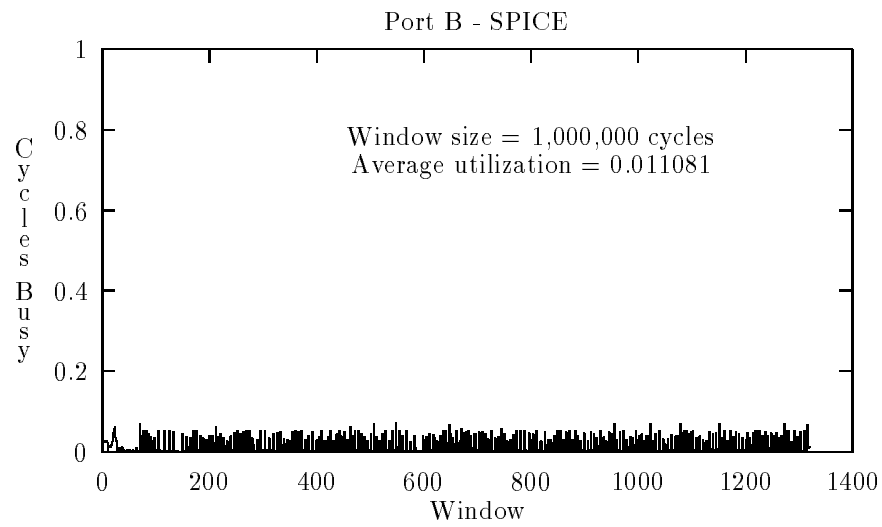
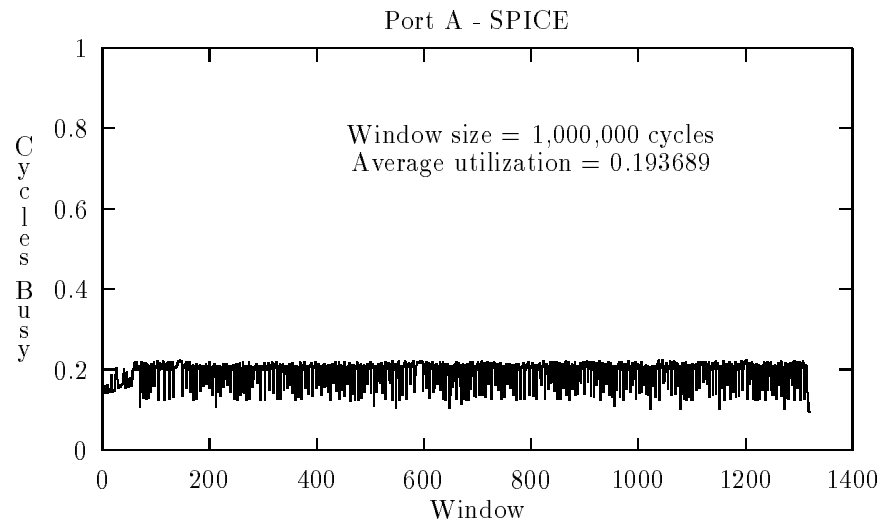


Figure 31: Port utilization for Perfect Benchmark SPICE.

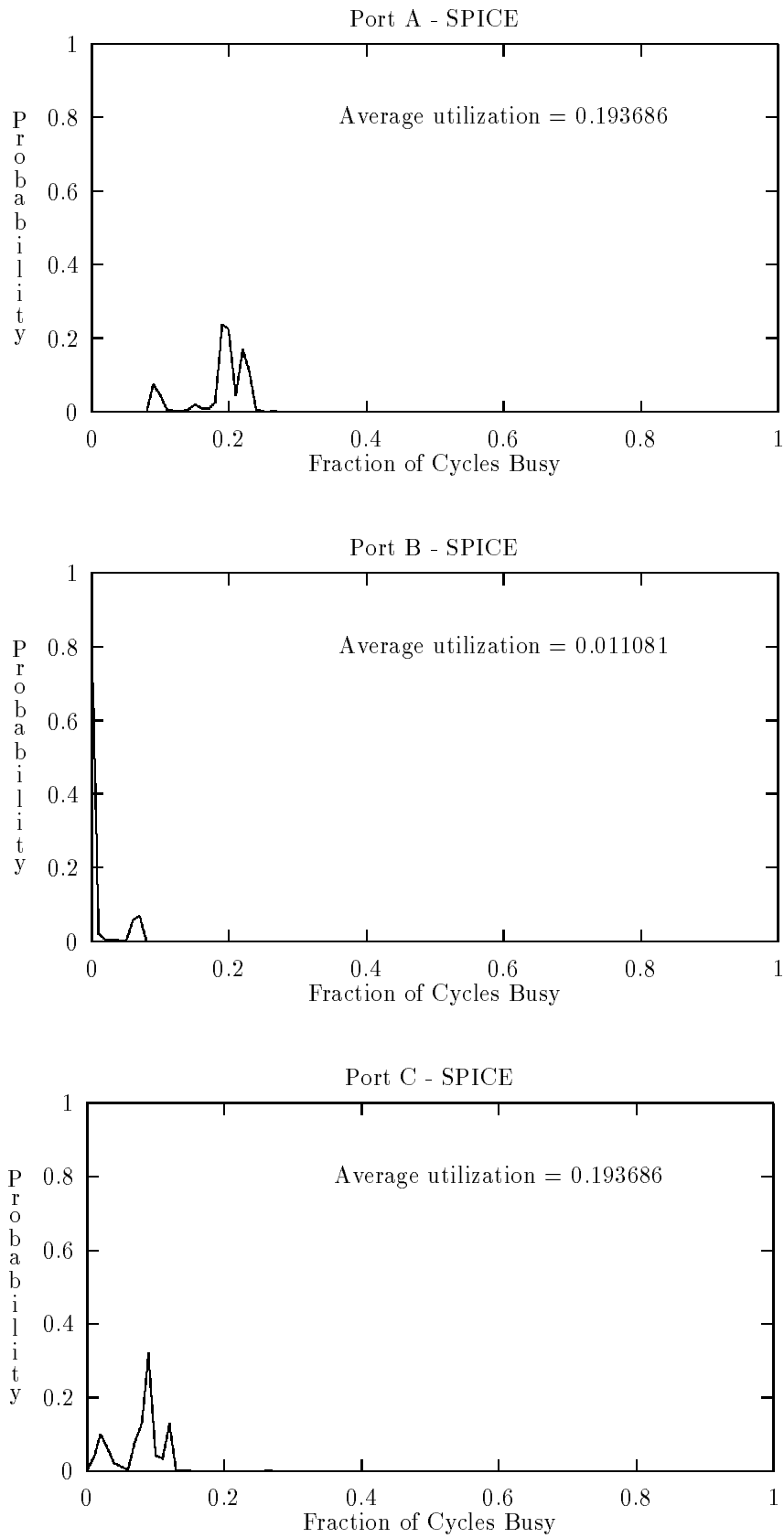


Figure 32: Port utilization histogram for Perfect Benchmark SPICE.

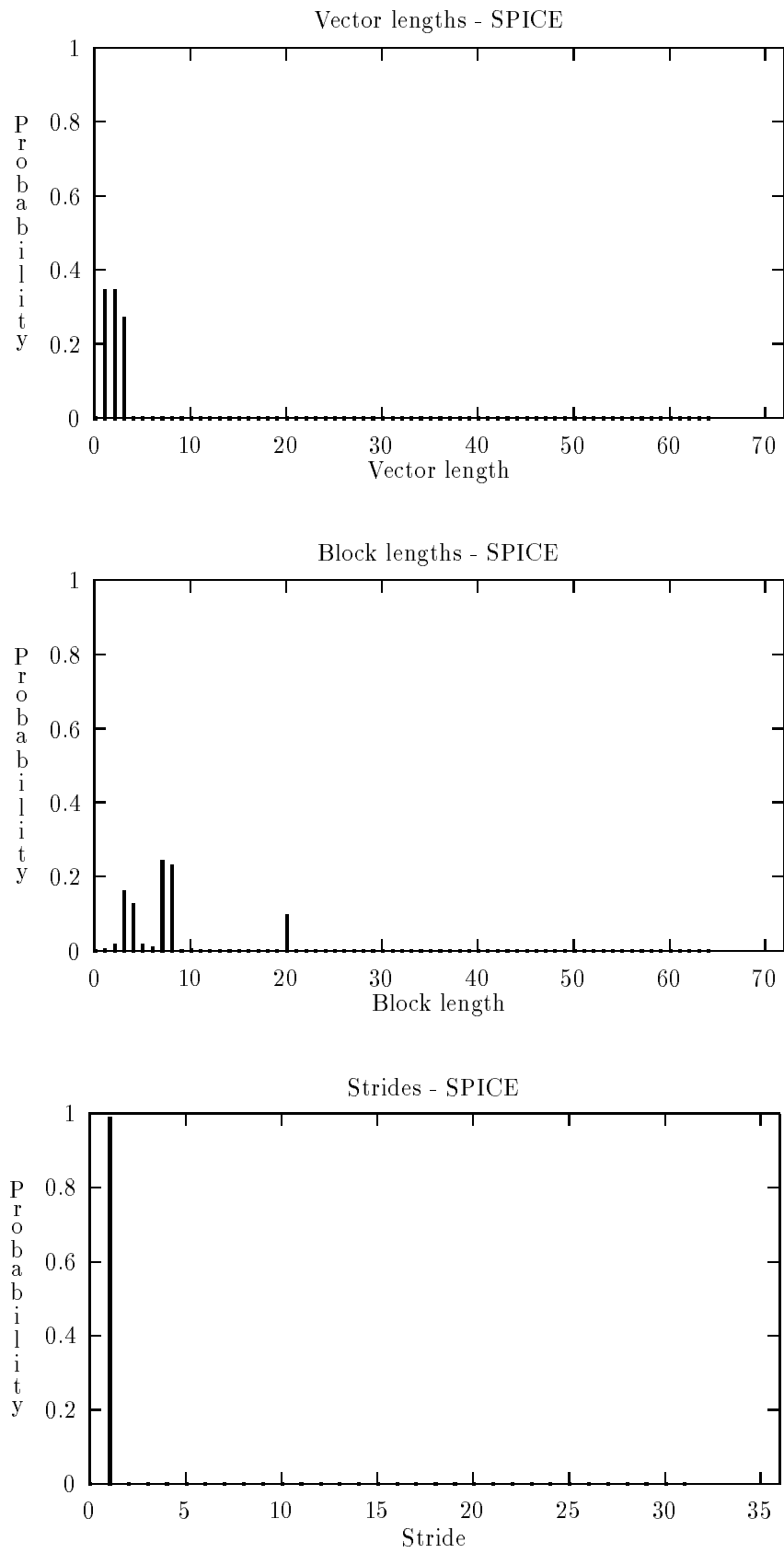


Figure 33: Distribution of lengths and strides for Perfect Benchmark SPICE.

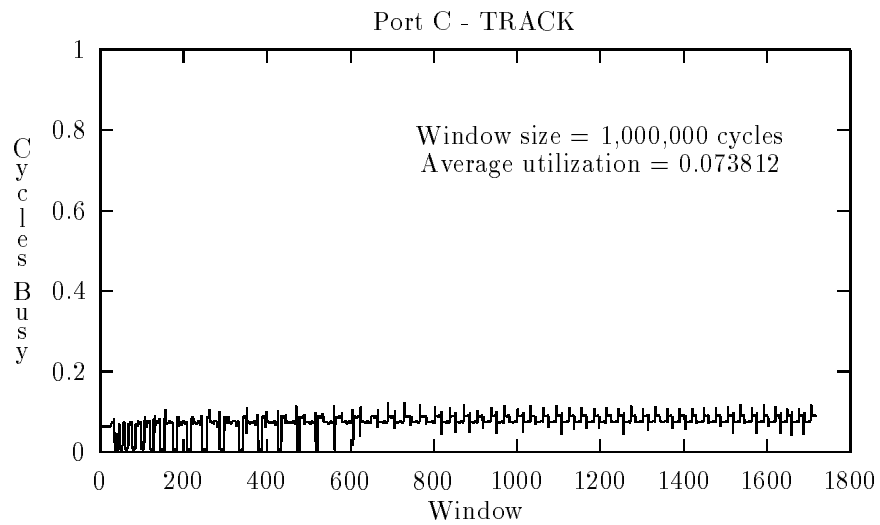
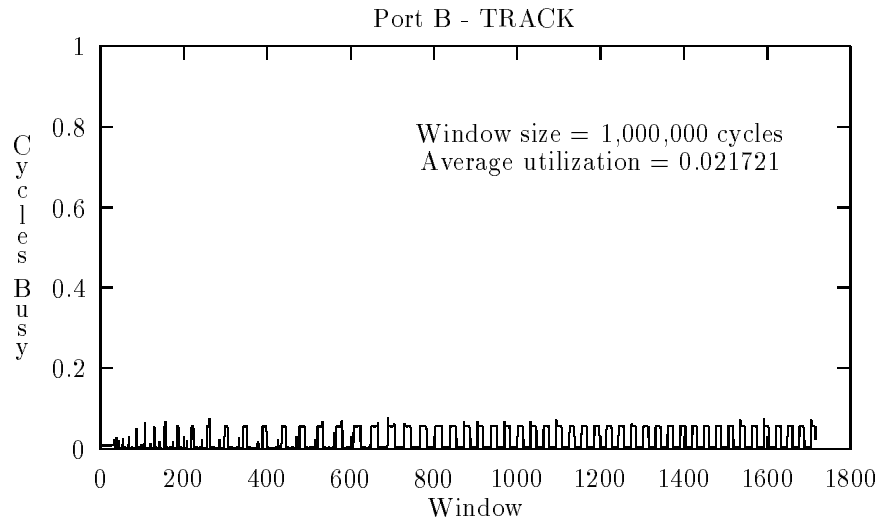
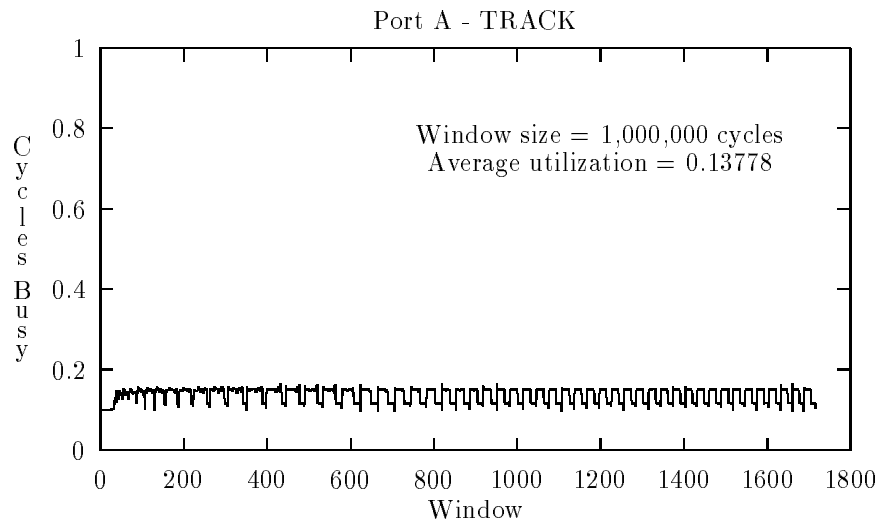


Figure 34: Port utilization for Perfect Benchmark TRACK.

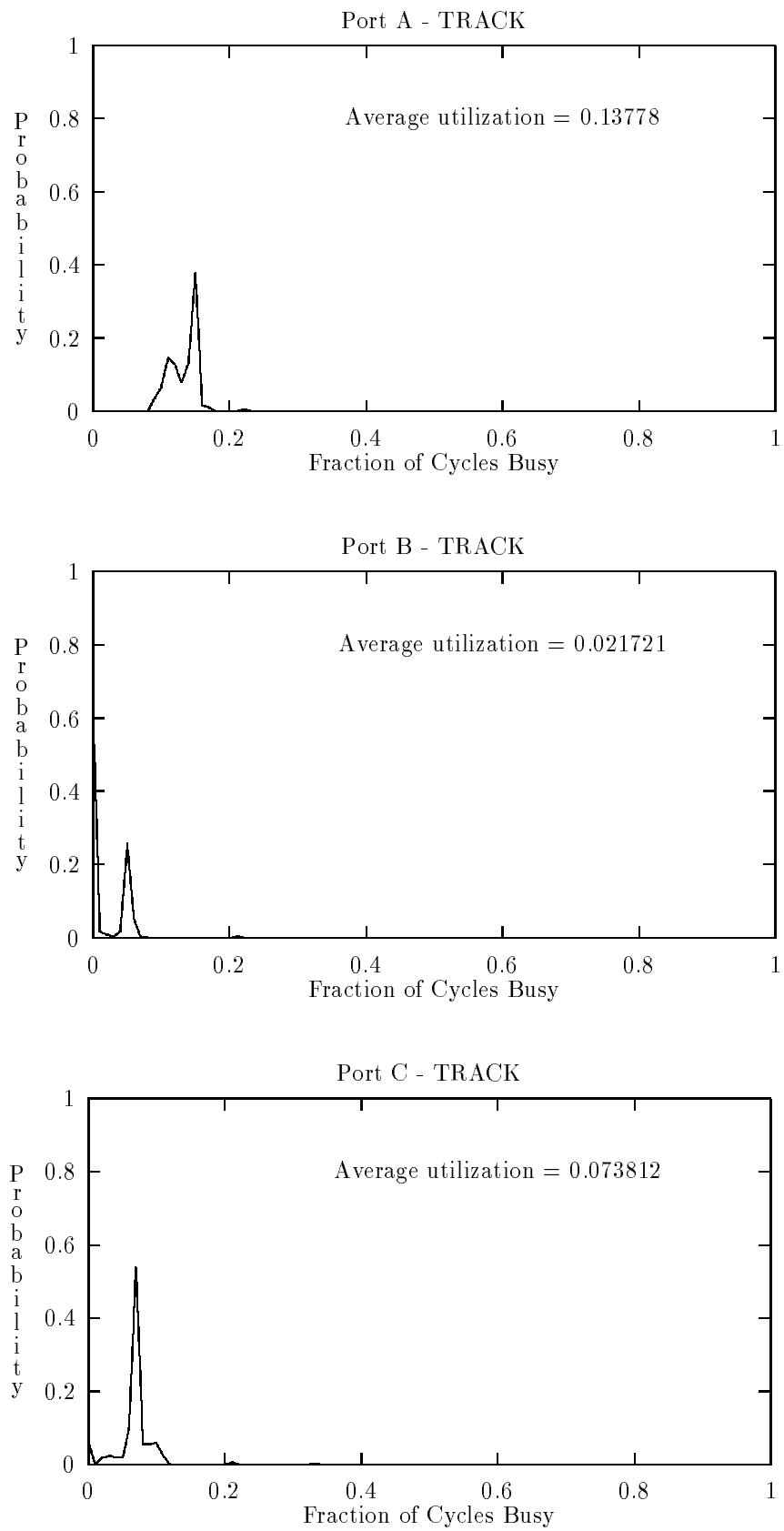


Figure 35: Port utilization histogram for Perfect Benchmark TRACK.

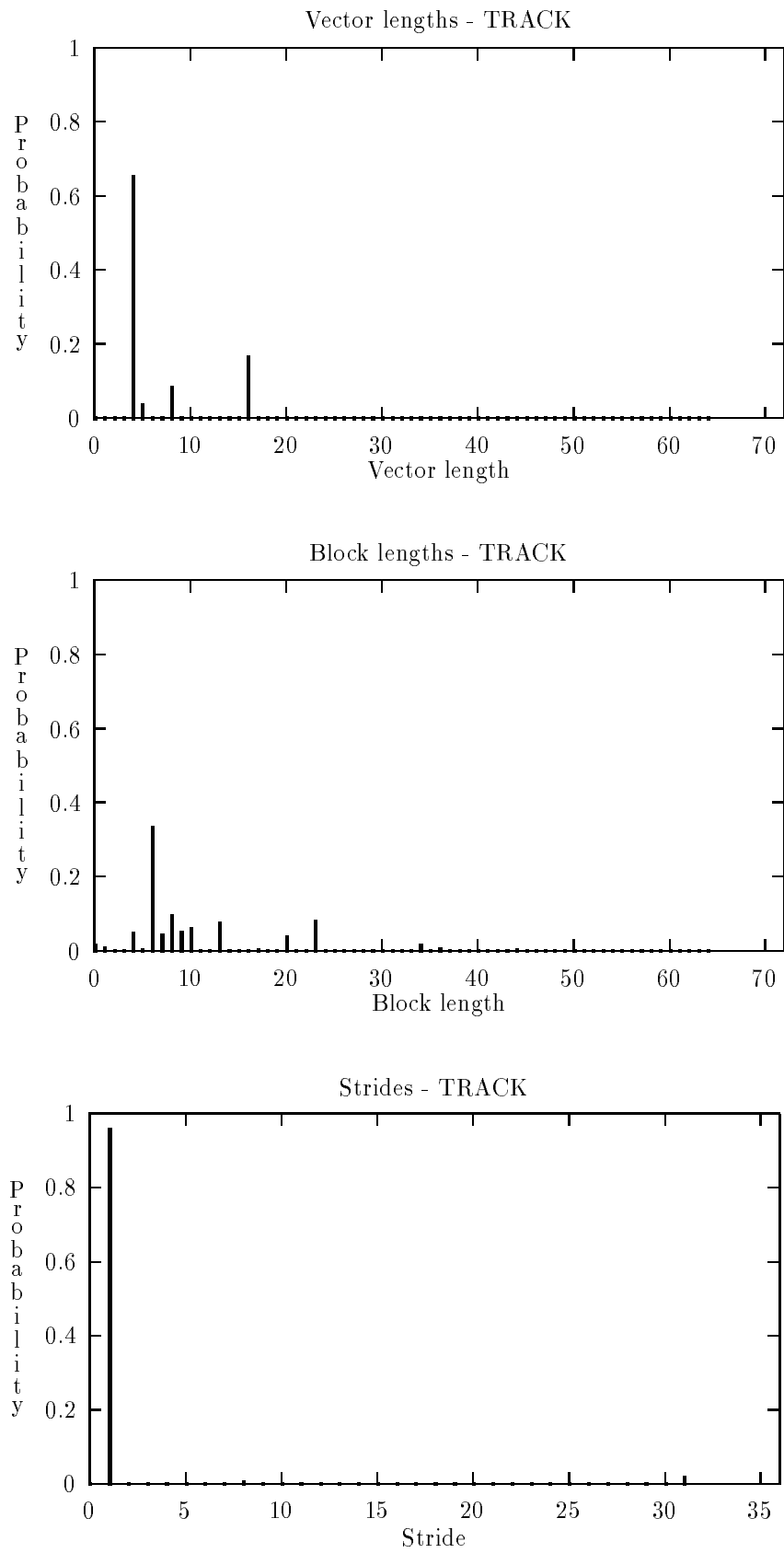


Figure 36: Distribution of lengths and strides for Perfect Benchmark TRACK.

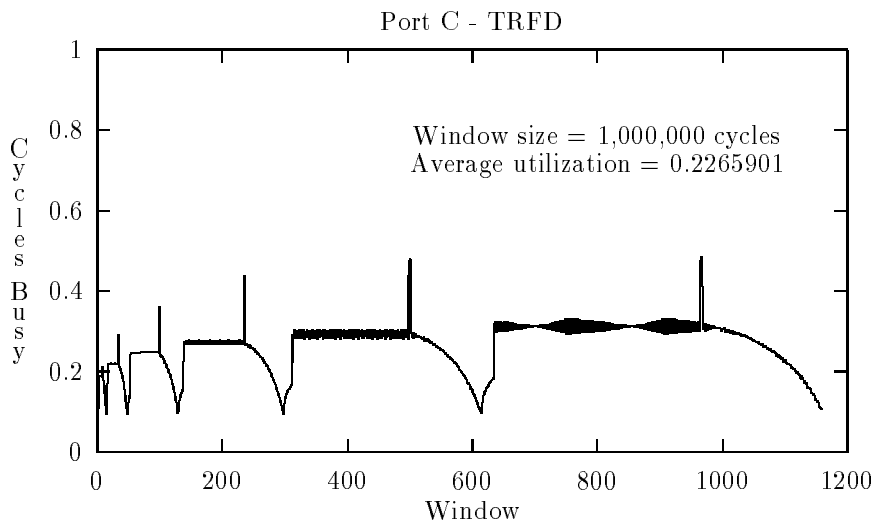
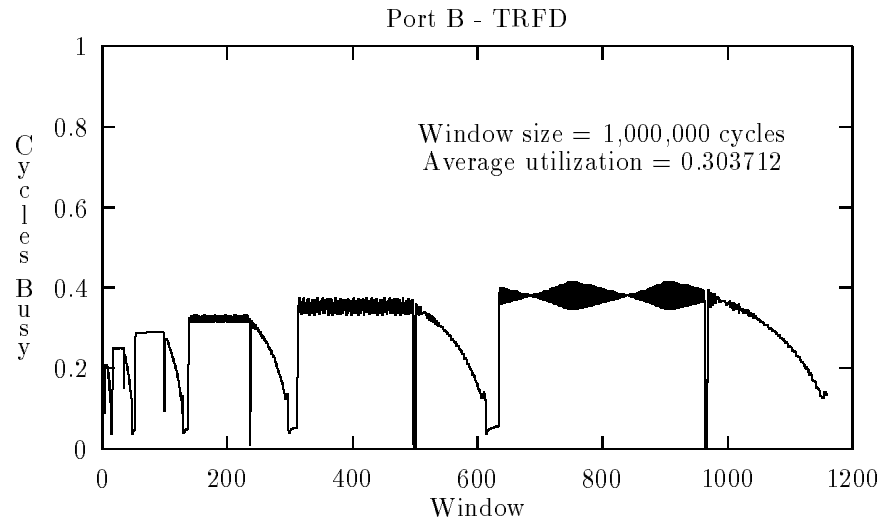
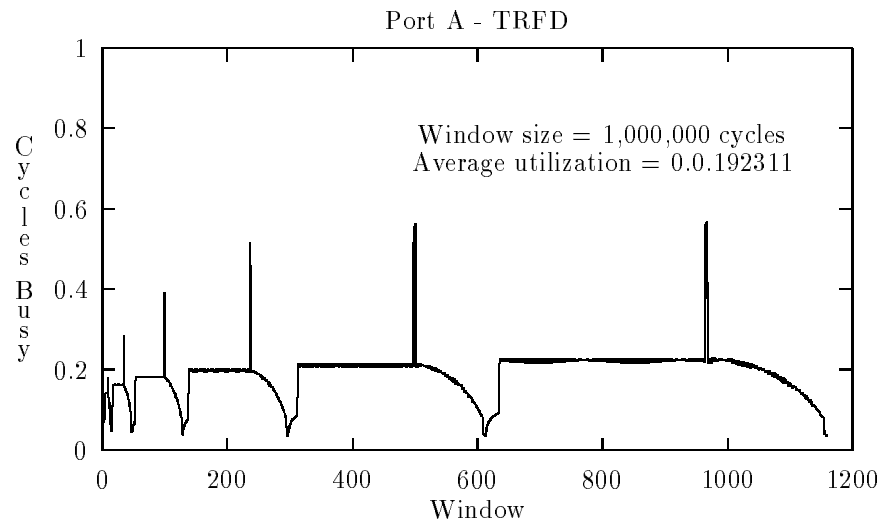


Figure 37: Port utilization for Perfect Benchmark TRFD.

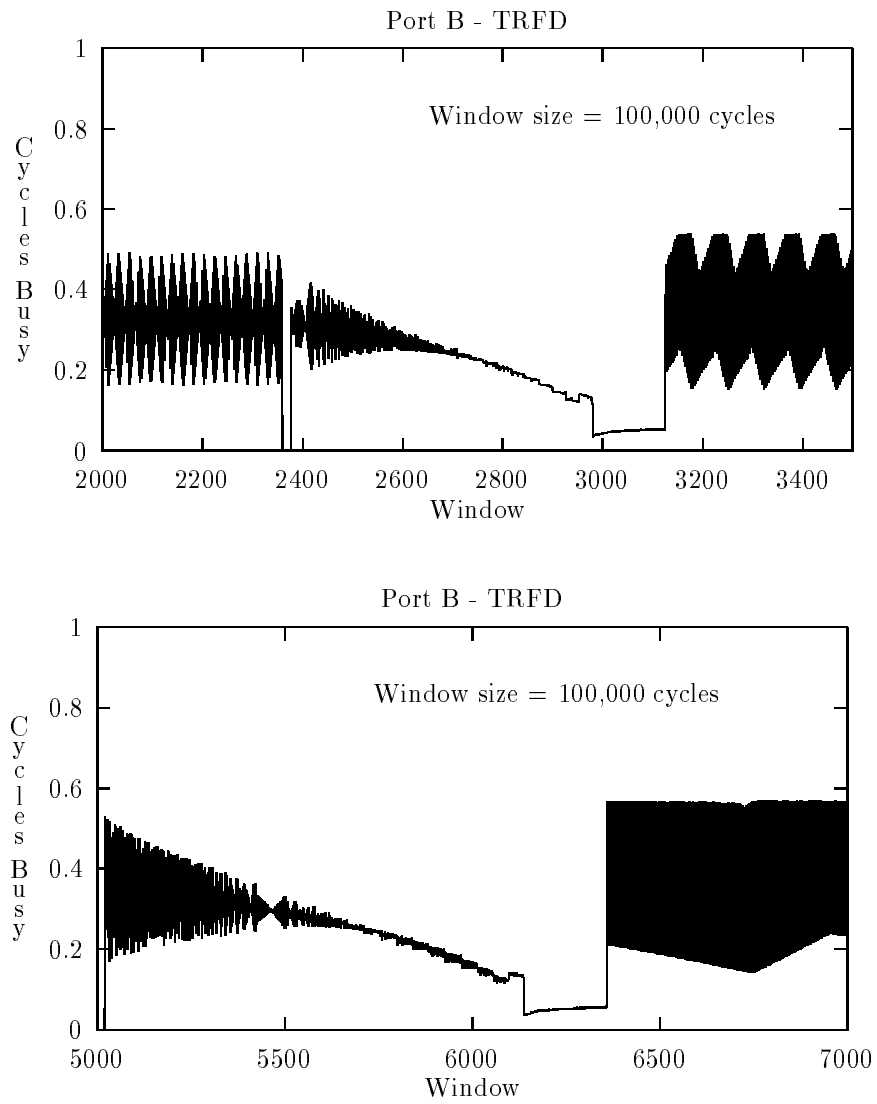


Figure 38: Port utilization for Perfect Benchmark TRFD. with a window size of 100,000

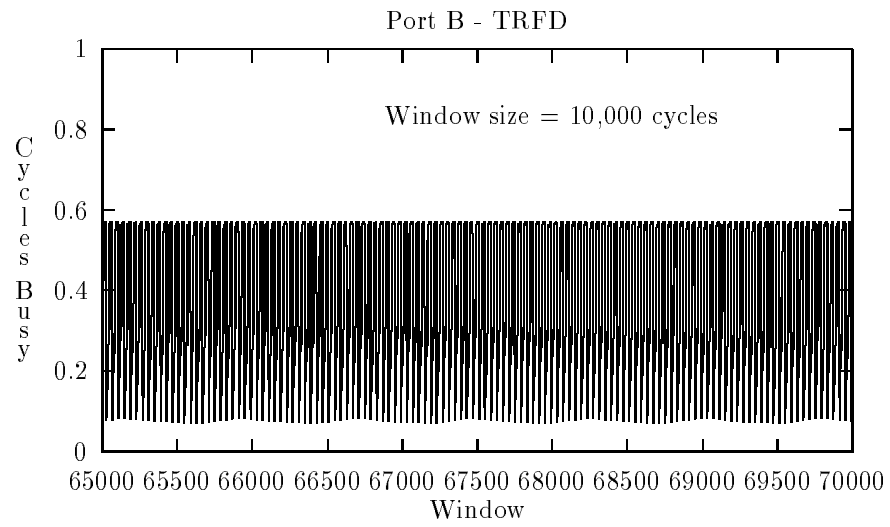


Figure 39: Port utilization for Perfect Benchmark TRFD. with a window size of 10,000

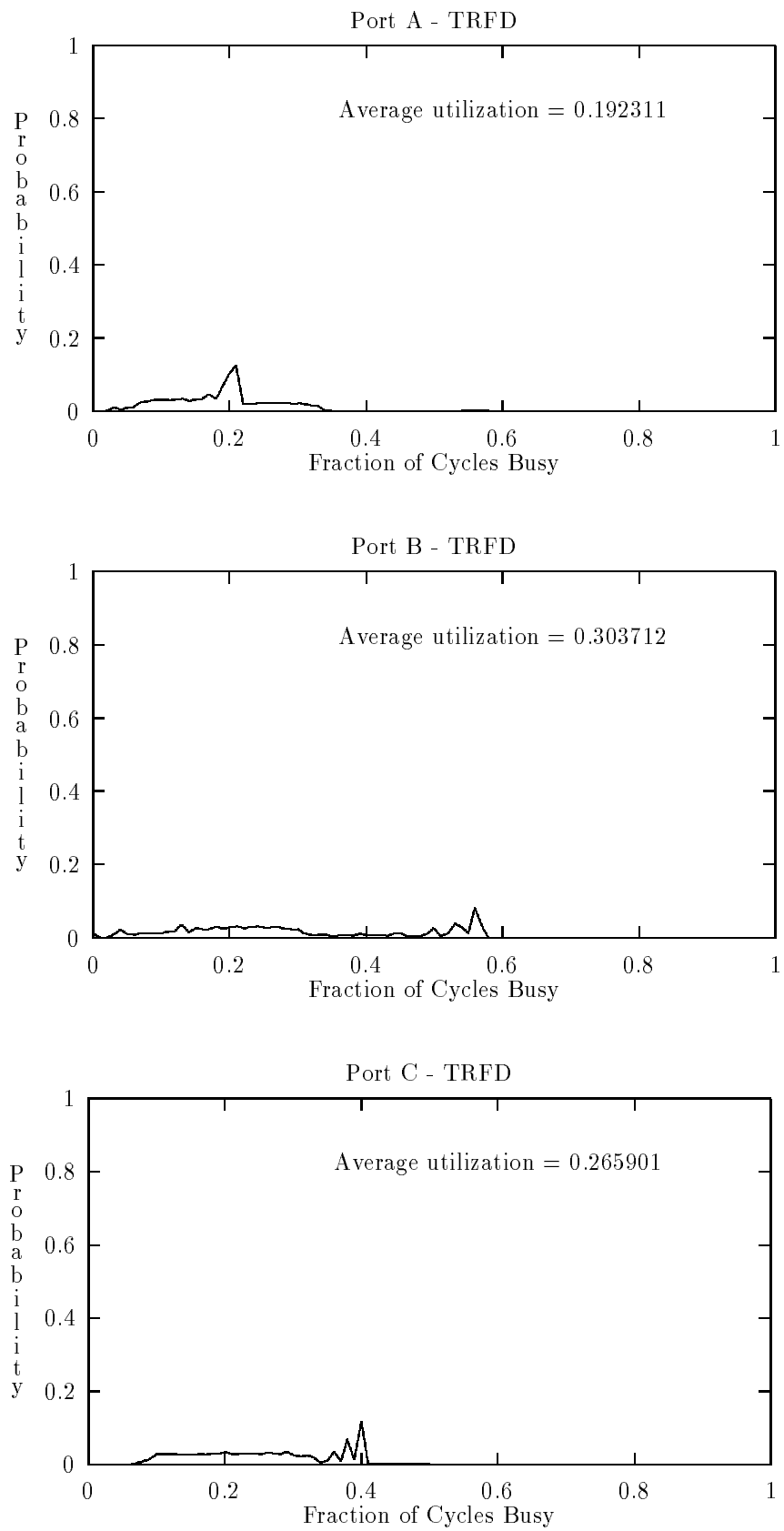


Figure 40: Port utilization histogram for Perfect Benchmark TRFD.

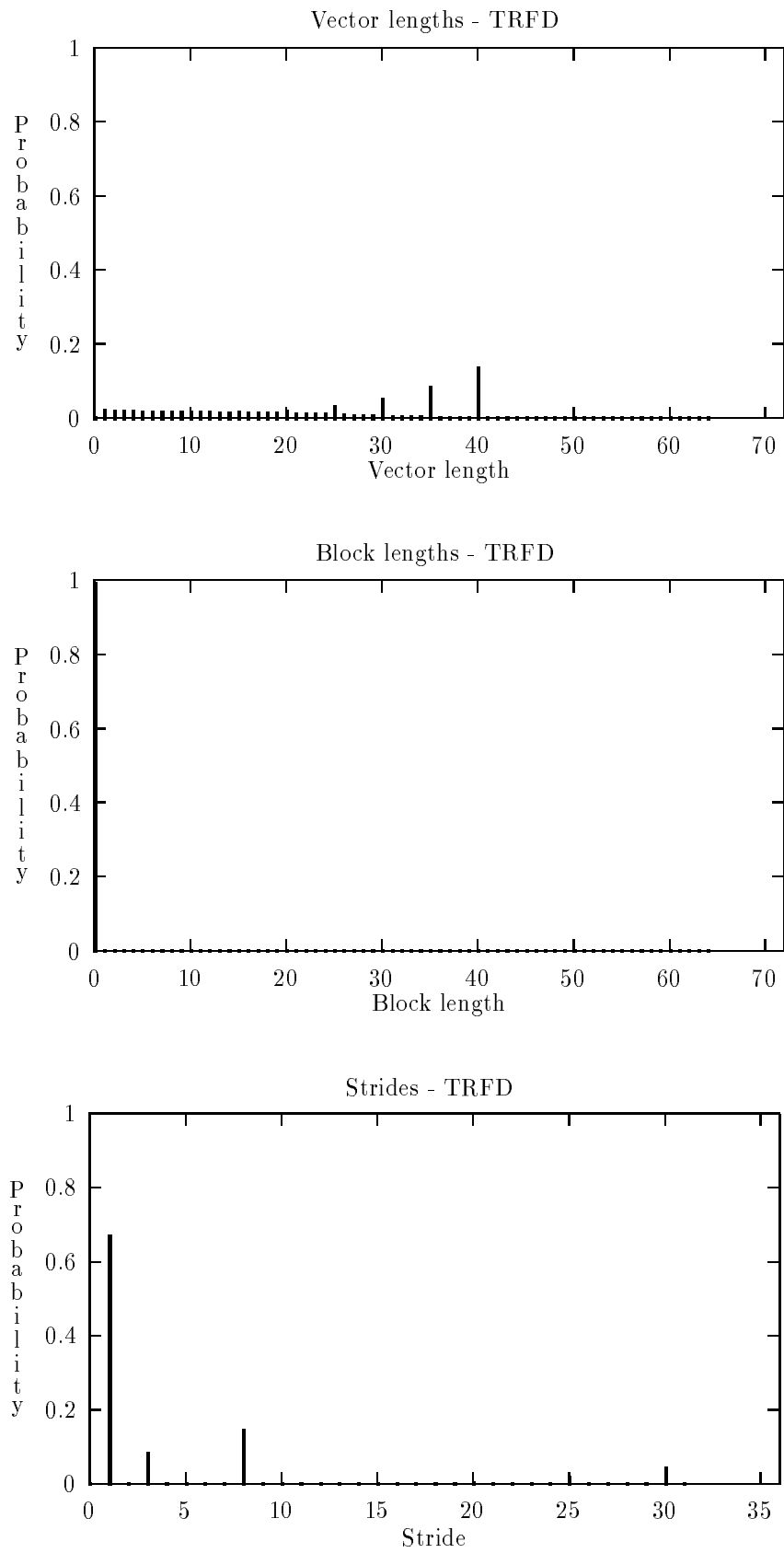


Figure 41: Distributions of lengths and strides for Perfect Benchmark TRFD.

A Hardware Performance Monitor Data

A.1 Statistics for ADM run on hpm

CP: 23.516s, Wallclock: 28.624s, 10.3% of 8-CPU Machine

```

Group 0: CPU seconds      :    23.52      CP executing      :    3919417233
-----
Million inst/sec (MIPS)  :    60.16      Instructions      :    1414698604
Avg. clock periods/inst :     2.77
% CP holding issue      :    49.27      CP holding issue  :    1931249636
Inst.buffer fetches/sec :     0.71M     Inst.buf. fetches:    166422229
Floating adds/sec       :    11.23M     F.P. adds        :    264080574
Floating multiplies/sec :    10.25M     F.P. multiplies  :    241130490
Floating reciprocal/sec :     0.98M     F.P. reciprocals :    23102715
I/O mem. references/sec :     0.01M     I/O references   :     127486
CPU mem. references/sec :    29.49M     CPU references   :    693440755
Floating ops/CPU second :    22.47M

```

```

Group 1: CPU seconds      :    23.51418  CP executing:    3919030713
-----

```

```

Hold issue condition      % of all CPs      actual # of CPs
Waiting on semaphores      :    0.00              2011
Waiting on shared registers :    0.00              1
Waiting on A-registers/funct. units:  9.93             389171577
Waiting on S-registers/funct. units: 17.11             670527087
Waiting on V-registers      :    9.18             359765983
Waiting on vector functional units : 11.04             432716373
Waiting on scalar memory references:  0.51             19911414
Waiting on block memory references :  4.28             167702464

```

```

Group 2: CPU seconds      :    23.49645  CP executing      :    3916075432
-----

```

```

Inst. buffer fetches/sec :     0.71M  total fetches      :    16642510
                               fetch conflicts :    24150517
I/O memory refs/sec      :     0.02M  actual refs        :     518533
  avg conflict/ref 1.25:          actual conflicts :     646939
Scalar memory refs/sec   :     4.63M  actual refs        :    108686562
Block memory refs/sec    :    24.89M  actual refs        :    584754192
CPU memory refs/sec      :    29.51M  actual refs        :    693440754
  avg conflict/ref 0.32:          actual conflicts :    222545696
CPU memory writes/sec    :    11.61M  actual refs        :    272713462
CPU memory reads/sec     :    17.91M  actual refs        :    420727292

```

```

Group 3: CPU seconds      :    23.48996  CP executing:    3914993318
-----

```

```

(octal) type of instruction  inst./CPUsec      actual inst.  % of all inst.
(000-017)jump/special       :    2.02M         47334870     3.35
(020-077)scalar functional unit :  48.53M        1139926821   80.58
(100-137)scalar memory      :    4.63M        108686562    7.68
(140-157,175)vector integer/log.:  0.19M         4469644      0.32
(160-174)vector floating point :  2.49M         58548267     4.14
(176-177)vector load and store :  2.37M         55732265     3.94

```

```

type of operation          ops/CPUsec      actual ops  avg. VL
Vector integer&logical     :    3.00M        70386327     15.75
Vector floating point      :    18.56M       435893834     7.45
Scalar functional unit     :    48.53M       1139926821

```


A.2 Statistics for ARC2D run on hpm

CP: 13.271s, Wallclock: 13.606s, 12.2% of 8-CPU Machine

```

Group 0: CPU seconds :      13.27      CP executing :      2211920114
-----
Million inst/sec (MIPS) :      30.16      Instructions :      400330327
Avg. clock periods/inst :       5.53
% CP holding issue :      79.42      CP holding issue :      1756655810
Inst.buffer fetches/sec :       0.03M      Inst.buf. fetches:      399165
Floating adds/sec :      60.84M      F.P. adds :      807504811
Floating multiplies/sec :     97.52M      F.P. multiplies :     1294277802
Floating reciprocal/sec :     10.00M      F.P. reciprocals :     132690070
I/O mem. references/sec :       0.02M      I/O references :      203008
CPU mem. references/sec :     145.99M      CPU references :     1937455492
Floating ops/CPU second :     168.37M

```

```

Group 1: CPU seconds :      13.27502 CP executing:      2212503899
-----

```

```

Hold issue condition          % of all CPs      actual # of CPs
Waiting on semaphores          : 0.00                1090
Waiting on shared registers    : 0.00                 1
Waiting on A-registers/funct. units: 5.42            120023696
Waiting on S-registers/funct. units: 1.59             35160771
Waiting on V-registers        : 25.36            561134233
Waiting on vector functional units : 45.43           1005048096
Waiting on scalar memory references: 0.87             19337223
Waiting on block memory references: 13.49            298386736

```

```

Group 2: CPU seconds :      13.27383      CP executing :      2212304228
-----

```

```

Inst. buffer fetches/sec :       0.03M      total fetches :      399070
                                          fetch conflicts :      812392
I/O memory refs/sec :       0.00M      actual refs :      1120
  avg conflict/ref 10.94:                actual conflicts :     12253
Scalar memory refs/sec :       0.17M      actual refs :     2288285
Block memory refs/sec :     145.79M      actual refs :     1935167179
CPU memory refs/sec :     145.96M      actual refs :     1937455464
  avg conflict/ref 0.05:                actual conflicts :     106038026
CPU memory writes/sec :       42.60M      actual refs :     565411335
CPU memory reads/sec :     103.36M      actual refs :     1372044129

```

```

Group 3: CPU seconds :      13.27455      CP executing:      2212425244
-----

```

```

(octal) type of instruction  inst./CPUsec      actual inst.  % of all inst.
(000-017)jump/special       :      1.25M      16652049      4.16
(020-077)scalar functional unit :     22.33M      296446625     74.05
(100-137)scalar memory      :       0.17M      2288343       0.57
(140-157,175)vector integer/log.:    0.37M      4943375       1.23
(160-174)vector floating point :     3.20M      42493466     10.61
(176-177)vector load and store :     2.83M      37506547      9.37

```

```

type of operation           ops/CPUsec      actual ops  avg. VL
Vector integer&logical      :     20.56M      272984145     55.22
Vector floating point       :     168.52M      2236984342     52.64
Scalar functional unit      :     22.33M      296446625

```

A.3 Statistics for BDNA run on hpm

CP: 13.271s, Wallclock: 13.606s, 12.2% of 8-CPU Machine

```

Group 0: CPU seconds      :      9.20      CP executing      :      1534119009
-----
Million inst/sec (MIPS) :      32.65      Instructions      :      300506668
Avg. clock periods/inst :       5.11
% CP holding issue      :      68.53      CP holding issue   :      1051375980
Inst.buffer fetches/sec :      0.43M      Inst.buf. fetches :      4001395
Floating adds/sec       :      61.63M      F.P. adds         :      567299194
Floating multiplies/sec :      56.69M      F.P. multiplies   :      521844022
Floating reciprocal/sec :      10.01M      F.P. reciprocals  :      92161734
I/O mem. references/sec :      0.00M      I/O references     :      16896
CPU mem. references/sec :      72.66M      CPU references     :      668835186
Floating ops/CPU second :      128.34M

```

```

Group 1: CPU seconds      :      9.20446  CP executing:      1534077321
-----

```

```

Hold issue condition      % of all CPs      actual # of CPs
Waiting on semaphores      :      0.00              2246
Waiting on shared registers :      0.00              1
Waiting on A-registers/funct. units:  4.44             68121830
Waiting on S-registers/funct. units:  8.84             135541642
Waiting on V-registers      :     19.01             291699024
Waiting on vector functional units :  37.09             569063937
Waiting on scalar memory references:  0.16              2380375
Waiting on block memory references :  3.84             58914341

```

```

Group 2: CPU seconds      :      9.20351  CP executing      :      1533917839
-----

```

```

Inst. buffer fetches/sec :      0.43M  total fetches      :      4001970
                               fetch conflicts :      4768923
I/O memory refs/sec      :      0.04M  actual refs        :      407586
  avg conflict/ref 0.43:      actual conflicts :      173653
Scalar memory refs/sec   :      2.26M  actual refs        :      20815289
Block memory refs/sec    :      70.41M  actual refs        :      648020036
CPU memory refs/sec      :      72.67M  actual refs        :      668835325
  avg conflict/ref 0.06:      actual conflicts :      42455622
CPU memory writes/sec    :      30.07M  actual refs        :      276714751
CPU memory reads/sec     :      42.61M  actual refs        :      392120574

```

```

Group 3: CPU seconds      :      9.20375  CP executing:      1533958779
-----

```

```

(octal) type of instruction  inst./CPUsec      actual inst.  % of all inst.
(000-017)jump/special      :      3.16M        29075414     9.68
(020-077)scalar functional unit :  22.38M        205993410    68.55
(100-137)scalar memory      :      2.26M        20815290     6.93
(140-157,175)vector integer/log.:  0.87M         7989252      2.66
(160-174)vector floating point :  2.52M        23209552     7.72
(176-177)vector load and store :  1.46M        13424370     4.47

```

```

type of operation          ops/CPUsec      actual ops  avg. VL
Vector integer&logical     :      37.18M     342202546   42.83
Vector floating point      :      128.29M    1180788820  50.88
Scalar functional unit     :      22.38M     205993410

```

A.4 Statistics for DYFESM run on hpm

CP: 12.173s, Wallclock: 12.484s, 12.2% of 8-CPU Machine

```

Group 0: CPU seconds      :      12.17      CP executing      :      2028951426
-----
Million inst/sec (MIPS) :      47.66      Instructions      :      580247408
Avg. clock periods/inst :       3.50
% CP holding issue      :      64.47      CP holding issue   :      1308110318
Inst.buffer fetches/sec :       0.11M      Inst.buf. fetches  :      1310723
Floating adds/sec       :      33.53M      F.P. adds         :      408209700
Floating multiplies/sec :      11.83M      F.P. multiplies   :      144026423
Floating reciprocal/sec :       0.00M      F.P. reciprocals  :       42547
I/O mem. references/sec :       0.02M      I/O references     :       284128
CPU mem. references/sec :      68.46M      CPU references     :      833452713
Floating ops/CPU second :      45.37M

```

```

Group 1: CPU seconds      :      12.17463  CP executing:      2029104402
-----

```

```

Hold issue condition      % of all CPs      actual # of CPs
Waiting on semaphores      :      0.00              1613
Waiting on shared registers :      0.00              1
Waiting on A-registers/funct. units:  6.44             130650148
Waiting on S-registers/funct. units: 12.16             246816665
Waiting on V-registers      :      18.31             371606170
Waiting on vector functional units :  10.00             202866278
Waiting on scalar memory references:  5.47             110900821
Waiting on block memory references :  13.20             267850064

```

```

Group 2: CPU seconds      :      12.17418  CP executing      :      2029030793
-----

```

```

Inst. buffer fetches/sec :       0.11M  total fetches      :      1313163
                                           fetch conflicts    :      1445788
I/O memory refs/sec      :       0.05M  actual refs        :      569863
  avg conflict/ref 0.28:                actual conflicts   :      158545
Scalar memory refs/sec   :       1.57M  actual refs        :      19056573
Block memory refs/sec    :      66.90M  actual refs        :      814396138
CPU memory refs/sec      :      68.46M  actual refs        :      833452711
  avg conflict/ref 0.06:                actual conflicts   :      49269133
CPU memory writes/sec    :      28.96M  actual refs        :      352616789
CPU memory reads/sec     :      39.50M  actual refs        :      480835922

```

```

Group 3: CPU seconds      :      12.17306  CP executing:      2028842593
-----

```

```

(octal) type of instruction  inst./CPUsec      actual inst.  % of all inst.
(000-017)jump/special        :      3.94M      47951925     8.26
(020-077)scalar functional unit :  36.53M      444717416    76.64
(100-137)scalar memory        :      1.57M      19056579     3.28
(140-157,175)vector integer/log.:  0.75M       9082145     1.57
(160-174)vector floating point :  2.38M      28942417     4.99
(176-177)vector load and store :  2.51M      30496927     5.26

```

```

type of operation           ops/CPUsec      actual ops  avg. VL
Vector integer&logical      :      25.16M     306306186   33.73
Vector floating point       :      42.92M     522454139   18.05
Scalar functional unit      :      36.53M     444717416

```

A.5 Statistics for FLO52 run on hpm

CP: 5.407s, Wallclock: 5.844s, 11.6% of 8-CPU Machine

```

Group 0: CPU seconds      :      5.41      CP executing      :      901368072
-----
Million inst/sec (MIPS) :      44.22      Instructions      :      239176324
Avg. clock periods/inst :       3.77
% CP holding issue      :      66.28      CP holding issue   :      597431466
Inst.buffer fetches/sec :       0.12M      Inst.buf. fetches :      625563
Floating adds/sec       :      58.84M      F.P. adds         :      318228128
Floating multiplies/sec :      53.97M      F.P. multiplies   :      291900026
Floating reciprocal/sec :       5.79M      F.P. reciprocals  :      31327679
I/O mem. references/sec :       0.20M      I/O references    :      1056898
CPU mem. references/sec :     132.56M      CPU references    :      716919580
Floating ops/CPU second :     118.61M

```

```

Group 1: CPU seconds      :      5.41366  CP executing:      902276509
-----

```

```

Hold issue condition      % of all CPs      actual # of CPs
Waiting on semaphores      :      0.00              759
Waiting on shared registers :      0.00              1
Waiting on A-registers/funct. units: 6.39              57617864
Waiting on S-registers/funct. units: 7.80              70369323
Waiting on V-registers      :     23.62             213107636
Waiting on vector functional units : 33.04             298127415
Waiting on scalar memory references: 0.52              4648730
Waiting on block memory references : 8.74              78864621

```

```

Group 2: CPU seconds      :      5.41098      CP executing      :      901830105
-----

```

```

Inst. buffer fetches/sec :       0.12M      total fetches      :      625465
                                       fetch conflicts   :      641064
I/O memory refs/sec      :       0.00M      actual refs        :      19223
  avg conflict/ref 0.69 :
Scalar memory refs/sec   :       1.05M      actual refs        :      5676225
Block memory refs/sec    :     131.44M      actual refs        :      711243355
CPU memory refs/sec      :     132.49M      actual refs        :      716919580
  avg conflict/ref 0.05 :
CPU memory writes/sec    :     36.47M      actual refs        :      197335727
CPU memory reads/sec     :     96.02M      actual refs        :      519583853

```

```

Group 3: CPU seconds      :      5.40916      CP executing:      901526711
-----

```

```

(octal) type of instruction  inst./CPUsec      actual inst.      % of all inst.
(000-017)jump/special      :      3.05M        16475030          6.89
(020-077)scalar functional unit : 33.89M        183332365         76.65
(100-137)scalar memory      :       1.05M        5676225           2.37
(140-157,175)vector integer/log.: 0.15M         789779            0.33
(160-174)vector floating point : 2.88M        15587640          6.52
(176-177)vector load and store : 3.20M        17315285          7.24

type of operation          ops/CPUsec      actual ops      avg. VL
Vector integer&logical     :      6.56M      35490636        44.94
Vector floating point      :     116.63M    630879683        40.47
Scalar functional unit     :      33.89M    183332365

```

A.6 Statistics for MDG run on hpm

CP: 172.882s, Wallclock: 422.178s, 5.1% of 8-CPU Machine

```

Group 0: CPU seconds      :    172.88      CP executing      :    28813813946
-----
Million inst/sec (MIPS)  :    49.29      Instructions      :    8521326842
Avg. clock periods/inst :     3.38
% CP holding issue      :    57.60      CP holding issue  :    16598151323
Inst.buffer fetches/sec :    0.50M     Inst.buf. fetches:    85787319
Floating adds/sec       :    12.35M     F.P. adds        :    2134362988
Floating multiplies/sec :    8.65M     F.P. multiplies  :    1495272369
Floating reciprocal/sec :    1.53M     F.P. reciprocals:    264938809
I/O mem. references/sec :    0.08M     I/O references   :    12974329
CPU mem. references/sec :    16.44M    CPU references   :    2842058903
Floating ops/CPU second :    22.53M

```

```

Group 1: CPU seconds      :    172.80337  CP executing:    28800561783
-----

```

```

Hold issue condition      % of all CPs      actual # of CPs
Waiting on semaphores     :    0.00              743
Waiting on shared registers :    0.00              1
Waiting on A-registers/funct. units: 3.75             1080153559
Waiting on S-registers/funct. units: 36.08            10391412772
Waiting on V-registers     :    5.98             1723500149
Waiting on vector functional units : 6.51             1876180187
Waiting on scalar memory references: 2.02             580492596
Waiting on block memory references : 0.90             258921443

```

```

Group 2: CPU seconds      :    172.94432  CP executing      :    28824053112
-----

```

```

Inst. buffer fetches/sec :    0.50M  total fetches      :    85788518
                               fetch conflicts :    167748691
I/O memory refs/sec      :    0.09M  actual refs        :    15811321
  avg conflict/ref 0.34:          actual conflicts :    5448854
Scalar memory refs/sec   :    6.24M  actual refs        :    1078963063
Block memory refs/sec    :    10.19M  actual refs        :    1763095840
CPU memory refs/sec      :    16.43M  actual refs        :    2842058903
  avg conflict/ref 0.07:          actual conflicts :    186039298
CPU memory writes/sec    :    6.31M  actual refs        :    1091626778
CPU memory reads/sec     :    10.12M  actual refs        :    1750432125

```

```

Group 3: CPU seconds      :    172.91009  CP executing:    28818347938
-----

```

```

(octal) type of instruction  inst./CPUsec      actual inst.  % of all inst.
(000-017)jump/special       :    2.48M        428399302    5.03
(020-077)scalar functional unit : 35.97M        6220010709   72.99
(100-137)scalar memory      :    6.24M        1078963120   12.66
(140-157,175)vector integer/log.: 1.32M         228337446    2.68
(160-174)vector floating point : 2.23M         385638037    4.53
(176-177)vector load and store : 1.04M         179978081    2.11

```

```

type of operation           ops/CPUsec      actual ops  avg. VL
Vector integer&logical      :    8.28M       1432433821   6.27
Vector floating point       :    15.14M     2617323003   6.79
Scalar functional unit      :    35.97M     6220010709

```

A.7 Statistics for MG3D run on hpm

CP: 124.798s, Wallclock: 251.524s, 6.2% of 8-CPU Machine

```

Group 0: CPU seconds      :    124.80      CP executing      :    20799762380
-----
Million inst/sec (MIPS)  :    48.53      Instructions      :    6056931402
Avg. clock periods/inst :     3.43
% CP holding issue      :    63.70      CP holding issue  :    13249531331
Inst.buffer fetches/sec :    0.29M     Inst.buf. fetches:    36121772
Floating adds/sec       :    42.68M     F.P. adds        :    5326749266
Floating multiplies/sec :    34.95M     F.P. multiplies  :    4362057943
Floating reciprocal/sec :     0.75M     F.P. reciprocals :    93368646
I/O mem. references/sec :     0.06M     I/O references   :    7370167
CPU mem. references/sec :    83.10M     CPU references    :    10370267982
Floating ops/CPU second :    78.38M

```

```

Group 1: CPU seconds      :    124.88585  CP executing:    20814308920
-----

```

```

Hold issue condition      % of all CPs      actual # of CPs
Waiting on semaphores     :     0.00          29058
Waiting on shared registers :     0.00           1
Waiting on A-registers/funct. units:  9.23          1922164153
Waiting on S-registers/funct. units:  7.00          1457392467
Waiting on V-registers    :    21.48          4470350475
Waiting on vector functional units :  24.84          5169755577
Waiting on scalar memory references:  0.34           71626084
Waiting on block memory references :  10.46          2176241435

```

```

Group 2: CPU seconds      :    125.06061  CP executing      :    20843434641
-----

```

```

Inst. buffer fetches/sec :    0.29M  total fetches      :    36127231
                               fetch conflicts :    84691584
I/O memory refs/sec      :    0.06M  actual refs        :    6910974
  avg conflict/ref 0.50:                actual conflicts :    3425025
Scalar memory refs/sec   :    3.17M  actual refs        :    396911854
Block memory refs/sec    :    79.75M  actual refs        :    9973356130
CPU memory refs/sec      :    82.92M  actual refs        :    10370267984
  avg conflict/ref 0.15:                actual conflicts :    1600724272
CPU memory writes/sec    :    34.64M  actual refs        :    4332639610
CPU memory reads/sec     :    48.28M  actual refs        :    6037628374

```

```

Group 3: CPU seconds      :    125.02146  CP executing:    20836909429
-----

```

```

(octal) type of instruction  inst./CPUsec      actual inst.  % of all inst.
(000-017)jump/special       :    1.20M        150192144    2.48
(020-077)scalar functional unit :    37.90M       4738635687   78.23
(100-137)scalar memory      :    3.17M        396911976    6.55
(140-157,175)vector integer/log.:  0.01M          703838       0.01
(160-174)vector floating point :    3.35M       418649888    6.91
(176-177)vector load and store :    2.81M       351837840    5.81

```

```

type of operation           ops/CPUsec      actual ops  avg. VL
Vector integer&logical      :    0.29M       36711844    52.16
Vector floating point       :    77.02M     9628979052   23.00
Scalar functional unit      :    37.90M     4738635687

```

A.8 Statistics for OCEAN run on hpm

CP: 20.899s, Wallclock: 20.970s, 12.5% of 8-CPU Machine

```

Group 0: CPU seconds      :    20.90      CP executing      :    3483370023
-----
Million inst/sec (MIPS)  :    45.91      Instructions      :    959507460
Avg. clock periods/inst :     3.63
% CP holding issue      :    62.04      CP holding issue  :    2160992887
Inst.buffer fetches/sec :     0.07M      Inst.buf. fetches:    1498388
Floating adds/sec       :    45.97M      F.P. adds        :    960839408
Floating multiplies/sec :    26.45M      F.P. multiplies  :    552770731
Floating reciprocal/sec :     0.91M      F.P. reciprocals :    18922813
I/O mem. references/sec :     0.00M      I/O references   :     76128
CPU mem. references/sec :    112.15M     CPU references    :    2343996516
Floating ops/CPU second :    73.33M

```

```

Group 1: CPU seconds      :    20.89814  CP executing:    3483022728
-----

```

```

Hold issue condition      % of all CPs      actual # of CPs
Waiting on semaphores     :     0.00              494
Waiting on shared registers :     0.00              1
Waiting on A-registers/funct. units:  3.82            133096969
Waiting on S-registers/funct. units: 12.63            439882137
Waiting on V-registers    :     9.70            337729863
Waiting on vector functional units : 18.43            641857508
Waiting on scalar memory references:  0.59            20382672
Waiting on block memory references : 18.51            644737250

```

```

Group 2: CPU seconds      :    20.89081  CP executing      :    3481801153
-----

```

```

Inst. buffer fetches/sec :     0.07M  total fetches      :    1502953
                               fetch conflicts :    1730562
I/O memory refs/sec      :     0.01M  actual refs        :    215775
  avg conflict/ref 0.51:                actual conflicts :    109627
Scalar memory refs/sec   :     8.21M  actual refs        :    171516299
Block memory refs/sec    :    103.99M  actual refs        :    2172480217
CPU memory refs/sec      :    112.20M  actual refs        :    2343996516
  avg conflict/ref 0.10:                actual conflicts :    223286023
CPU memory writes/sec    :    54.54M  actual refs        :    1139340216
CPU memory reads/sec     :    57.66M  actual refs        :    1204656300

```

```

Group 3: CPU seconds      :    20.89906  CP executing:    3483177348
-----

```

```

(octal) type of instruction  inst./CPUsec      actual inst.  % of all inst.
(000-017)jump/special       :     2.18M        45594593     4.75
(020-077)scalar functional unit :  31.89M        666381021    69.45
(100-137)scalar memory      :     8.21M        171516300    17.88
(140-157,175)vector integer/log.:  0.01M          258991       0.03
(160-174)vector floating point :  1.50M          31356126     3.27
(176-177)vector load and store :  2.12M          44400448     4.63

```

```

type of operation           ops/CPUsec      actual ops  avg. VL
Vector integer&logical      :     0.55M      11430469    44.13
Vector floating point       :     66.54M     1390688694  44.35
Scalar functional unit      :     31.89M      666381021

```

A.9 Statistics for QCD run on hpm

CP: 19.772s, Wallclock: 19.924s, 12.4% of 8-CPU Machine

```

Group 0: CPU seconds      :    19.77      CP executing      :    3295396717
-----
Million inst/sec (MIPS)  :    62.43      Instructions      :    1234302277
Avg. clock periods/inst :     2.67
% CP holding issue      :    43.00      CP holding issue  :    1417033901
Inst.buffer fetches/sec :    0.79M     Inst.buf. fetches:    15583525
Floating adds/sec       :    6.37M     F.P. adds        :    125880329
Floating multiplies/sec :    6.37M     F.P. multiplies  :    125893541
Floating reciprocal/sec :    0.15M     F.P. reciprocals :     3041047
I/O mem. references/sec :    0.03M     I/O references   :     509890
CPU mem. references/sec :    16.34M    CPU references   :    323047717
Floating ops/CPU second :    12.89M

```

```

Group 1: CPU seconds      :    19.77272  CP executing:    3295452651
-----

```

```

Hold issue condition      % of all CPs      actual # of CPs
Waiting on semaphores      :    0.00              650
Waiting on shared registers :    0.00              1
Waiting on A-registers/funct. units:  1.82             60011477
Waiting on S-registers/funct. units: 37.01            1219781905
Waiting on V-registers      :    0.10              3435173
Waiting on vector functional units :  0.01              203843
Waiting on scalar memory references:  1.27             41784834
Waiting on block memory references:  1.33             43961638

```

```

Group 2: CPU seconds      :    19.77143  CP executing      :    3295237888
-----

```

```

Inst. buffer fetches/sec :    0.79M  total fetches      :    15583384
                               fetch conflicts :    15730744
I/O memory refs/sec      :    0.01M  actual refs        :    174781
  avg conflict/ref 2.29:    0.01M  actual conflicts   :    399576
Scalar memory refs/sec   :    6.72M  actual refs        :    132819536
Block memory refs/sec    :    9.62M  actual refs        :    190228181
CPU memory refs/sec      :    16.34M  actual refs        :    323047717
  avg conflict/ref 0.05:    16.34M  actual conflicts   :    17502993
CPU memory writes/sec    :    6.86M  actual refs        :    135561566
CPU memory reads/sec     :    9.48M  actual refs        :    187486151

```

```

Group 3: CPU seconds      :    19.77377  CP executing:    3295627836
-----

```

```

(octal) type of instruction  inst./CPUsec      actual inst.  % of all inst.
(000-017)jump/special       :    2.68M        52958426     4.29
(020-077)scalar functional unit :  52.80M       1044052047   84.59
(100-137)scalar memory      :    6.72M       132819536   10.76
(140-157,175)vector integer/log.:  0.00M         61589        0.00
(160-174)vector floating point :  0.01M        196973        0.02
(176-177)vector load and store :  0.21M       4213714        0.34

type of operation            ops/CPUsec      actual ops  avg. VL
Vector integer&logical       :    0.06M       1274353     20.69
Vector floating point        :    0.18M       3544090     17.99
Scalar functional unit       :    52.80M     1044052047

```


A.10 Statistics for SPEC77 run on hpm

CP: 54.139s, Wallclock: 90.063s, 7.5% of 8-CPU Machine

| | | | | | |
|-------------------------|---|--------|--------------------|---|------------|
| Group 0: CPU seconds | : | 54.14 | CP executing | : | 9023342196 |
| ----- | | | | | |
| Million inst/sec (MIPS) | : | 53.53 | Instructions | : | 2897957566 |
| Avg. clock periods/inst | : | 3.11 | | | |
| % CP holding issue | : | 56.51 | CP holding issue | : | 5098896464 |
| Inst.buffer fetches/sec | : | 0.40M | Inst.buf. fetches: | : | 21548024 |
| Floating adds/sec | : | 20.82M | F.P. adds | : | 1127168781 |
| Floating multiplies/sec | : | 15.68M | F.P. multiplies | : | 848738252 |
| Floating reciprocal/sec | : | 0.27M | F.P. reciprocals | : | 14675640 |
| I/O mem. references/sec | : | 0.07M | I/O references | : | 3811562 |
| CPU mem. references/sec | : | 33.18M | CPU references | : | 1796577187 |
| Floating ops/CPU second | : | 36.77M | | | |

| | | | | | |
|----------------------|---|----------|---------------|---|------------|
| Group 1: CPU seconds | : | 54.07864 | CP executing: | : | 9013107192 |
| ----- | | | | | |

| Hold issue condition | % of all CPs | actual # of CPs |
|--------------------------------------|--------------|-----------------|
| Waiting on semaphores | : 0.00 | 2829 |
| Waiting on shared registers | : 0.00 | 1 |
| Waiting on A-registers/funct. units: | 6.62 | 596288333 |
| Waiting on S-registers/funct. units: | 10.65 | 960298731 |
| Waiting on V-registers | : 16.47 | 1484652293 |
| Waiting on vector functional units | : 14.73 | 1327726342 |
| Waiting on block memory references | : 8.88 | 800517972 |
| Waiting on scalar memory references: | 1.20 | 107861301 |

| | | | | | |
|----------------------|---|----------|--------------|---|------------|
| Group 2: CPU seconds | : | 54.05767 | CP executing | : | 9009611718 |
| ----- | | | | | |

| | | | | | |
|--------------------------|---|--------|------------------|---|------------|
| Inst. buffer fetches/sec | : | 0.40M | total fetches | : | 21540079 |
| | | | fetch conflicts | : | 31417455 |
| I/O memory refs/sec | : | 0.06M | actual refs | : | 3170928 |
| avg conflict/ref 0.70: | | | actual conflicts | : | 2226703 |
| Scalar memory refs/sec | : | 2.00M | actual refs | : | 107891690 |
| Block memory refs/sec | : | 31.24M | actual refs | : | 1688685240 |
| CPU memory refs/sec | : | 33.23M | actual refs | : | 1796576930 |
| avg conflict/ref 1.14: | | | actual conflicts | : | 2044923190 |
| CPU memory writes/sec | : | 9.97M | actual refs | : | 538881367 |
| CPU memory reads/sec | : | 23.27M | actual refs | : | 1257695563 |

| | | | | | |
|----------------------|---|----------|---------------|---|------------|
| Group 3: CPU seconds | : | 54.10629 | CP executing: | : | 9017715265 |
| ----- | | | | | |

| (octal) type of instruction | inst./CPUsec | actual inst. | % of all inst. |
|-----------------------------------|--------------|--------------|----------------|
| (000-017)jump/special | : 3.45M | 186482090 | 6.43 |
| (020-077)scalar functional unit | : 42.41M | 2294377378 | 79.17 |
| (100-137)scalar memory | : 1.99M | 107891690 | 3.72 |
| (140-157,175)vector integer/log.: | 0.48M | 25916186 | 0.89 |
| (160-174)vector floating point | : 2.79M | 151115374 | 5.21 |
| (176-177)vector load and store | : 2.44M | 132174898 | 4.56 |

| type of operation | ops/CPUsec | actual ops | avg. VL |
|------------------------|------------|------------|---------|
| Vector integer&logical | : 7.03M | 380626364 | 14.69 |
| Vector floating point | : 34.45M | 1864116674 | 12.34 |
| Scalar functional unit | : 42.41M | 2294377378 | |

A.11 Statistics for SPICE run on hpm

CP: 8.177s, Wallclock: 8.531s, 12.0% of 8-CPU Machine

```

Group 0: CPU seconds      :      8.18      CP executing      :      1363014389
-----
Million inst/sec (MIPS) :      40.04      Instructions      :      327488723
Avg. clock periods/inst :       4.16
% CP holding issue      :      56.52      CP holding issue   :      770386549
Inst.buffer fetches/sec :      0.74M      Inst.buf. fetches :      6090017
Floating adds/sec       :      3.56M      F.P. adds         :      29088101
Floating multiplies/sec :      3.03M      F.P. multiplies   :      24755187
Floating reciprocal/sec :      0.45M      F.P. reciprocals  :      3662453
I/O mem. references/sec :      0.06M      I/O references    :      480698
CPU mem. references/sec :      12.32M     CPU references     :      100732254
Floating ops/CPU second :      7.03M

```

```

Group 1: CPU seconds      :      8.17724  CP executing:      1362873882
-----

```

```

Hold issue condition      % of all CPs      actual # of CPs
Waiting on semaphores     :      0.00              931
Waiting on shared registers :      0.00              1
Waiting on A-registers/funct. units: 6.98              95116608
Waiting on S-registers/funct. units: 47.00             640594862
Waiting on V-registers    :      1.11             15121122
Waiting on vector functional units : 0.57              7801267
Waiting on scalar memory references: 0.34              4578883
Waiting on block memory references : 0.22             2965737

```

```

Group 2: CPU seconds      :      8.17904  CP executing      :      1363172790
-----

```

```

Inst. buffer fetches/sec :      0.74M      total fetches      :      6090668
                               fetch conflicts :      8683854
I/O memory refs/sec      :      0.08M      actual refs        :      676459
  avg conflict/ref 0.45 :              actual conflicts :      305675
Scalar memory refs/sec   :      10.42M     actual refs        :      85219842
Block memory refs/sec    :      1.90M      actual refs        :      15512433
CPU memory refs/sec      :      12.32M     actual refs        :      100732275
  avg conflict/ref 0.11 :              actual conflicts :      11436570
CPU memory writes/sec    :      3.57M      actual refs        :      29187597
CPU memory reads/sec     :      8.75M      actual refs        :      71544678

```

```

Group 3: CPU seconds      :      8.17919  CP executing:      1363197937
-----

```

```

(octal) type of instruction  inst./CPUsec      actual inst.      % of all inst.
(000-017)jump/special       :      3.26M        26653713         8.14
(020-077)scalar functional unit : 25.78M        210877291        64.39
(100-137)scalar memory      :      10.42M        85219845         26.02
(140-157,175)vector integer/log.: 0.01M         42811            0.01
(160-174)vector floating point : 0.26M        2129314          0.65
(176-177)vector load and store : 0.31M        2565750          0.78

```

```

type of operation           ops/CPUsec      actual ops      avg. VL
Vector integer&logical      :      0.05M        400732          9.36
Vector floating point       :      0.52M        4264149         2.00
Scalar functional unit      :      25.78M        210877291

```

A.12 Statistics for TRACK run on hpm

CP: 10.180s, Wallclock: 10.407s, 12.2% of 8-CPU Machine

```

Group 0: CPU seconds : 10.18 CP executing : 1696872192
-----
Million inst/sec (MIPS) : 48.82 Instructions : 497083627
Avg. clock periods/inst : 3.41
% CP holding issue : 50.38 CP holding issue : 854888740
Inst.buffer fetches/sec : 0.39M Inst.buf. fetches: 3975803
Floating adds/sec : 3.91M F.P. adds : 39775423
Floating multiplies/sec : 4.28M F.P. multiplies : 43596291
Floating reciprocal/sec : 0.13M F.P. reciprocals : 1360244
I/O mem. references/sec : 0.00M I/O references : 25750
CPU mem. references/sec : 12.15M CPU references : 123693290
Floating ops/CPU second : 8.32M

```

```

Group 1: CPU seconds : 10.18087 CP executing: 1696812371
-----

```

```

Hold issue condition      % of all CPs      actual # of CPs
Waiting on semaphores      : 0.00              883
Waiting on shared registers : 0.00              1
Waiting on A-registers/funct. units: 2.24             37969779
Waiting on S-registers/funct. units: 44.94             762485347
Waiting on V-registers      : 0.26             4349312
Waiting on vector functional units : 0.40             6809117
Waiting on scalar memory references: 0.77             13118717
Waiting on block memory references : 0.65             11055532

```

```

Group 2: CPU seconds : 10.17881 CP executing : 1696469059
-----

```

```

Inst. buffer fetches/sec : 0.39M total fetches : 3978051
                                fetch conflicts : 4448282
I/O memory refs/sec : 0.16M actual refs : 1582073
    avg conflict/ref 0.12: actual conflicts : 190620
Scalar memory refs/sec : 6.64M actual refs : 67593929
Block memory refs/sec : 5.51M actual refs : 56099361
CPU memory refs/sec : 12.15M actual refs : 123693290
    avg conflict/ref 0.05: actual conflicts : 5818890
CPU memory writes/sec : 4.17M actual refs : 42479766
CPU memory reads/sec : 7.98M actual refs : 81213524

```

```

Group 3: CPU seconds : 10.18005 CP executing: 1696675534
-----

```

```

(octal) type of instruction  inst./CPUsec      actual inst.  % of all inst.
(000-017)jump/special      : 4.37M           44456586     8.94
(020-077)scalar functional unit : 37.42M          380970399    76.64
(100-137)scalar memory      : 6.64M           67593993     13.60
(140-157,175)vector integer/log.: 0.02M           228801        0.05
(160-174)vector floating point : 0.13M           1330307       0.27
(176-177)vector load and store : 0.25M           2503612       0.50

type of operation          ops/CPUsec      actual ops    avg. VL
Vector integer&logical     : 0.25M          2517996       11.01
Vector floating point      : 0.69M          7068515        5.31
Scalar functional unit     : 37.42M         380970399

```

A.13 Statistics for TRFD run on hpm

CP: 7.915s, Wallclock: 8.237s, 12.0% of 8-CPU Machine

```

Group 0: CPU seconds      :      7.92      CP executing      :      1319213780
-----
Million inst/sec (MIPS) :      66.30      Instructions      :      524801122
Avg. clock periods/inst :       2.51
% CP holding issue      :      52.81      CP holding issue   :      696645376
Inst.buffer fetches/sec :      0.04M      Inst.buf. fetches :      300471
Floating adds/sec       :      27.36M      F.P. adds         :      216585227
Floating multiplies/sec :      27.34M      F.P. multiplies   :      216383479
Floating reciprocal/sec :      0.09M      F.P. reciprocals  :      735440
I/O mem. references/sec :      0.06M      I/O references    :      467121
CPU mem. references/sec :      84.64M      CPU references     :      669917794
Floating ops/CPU second :      54.79M
  
```

```

Group 1: CPU seconds      :      7.92758  CP executing:      1321262933
-----
  
```

```

Hold issue condition      % of all CPs      actual # of CPs
Waiting on semaphores      :      0.00              727
Waiting on shared registers :      0.00              1
Waiting on A-registers/funct. units:  8.80             116225516
Waiting on S-registers/funct. units: 14.33             189316362
Waiting on V-registers      :      0.97             12873619
Waiting on vector functional units :  0.91             12010702
Waiting on scalar memory references: 10.68             141087446
Waiting on block memory references : 16.46             217506481
  
```

```

Group 2: CPU seconds      :      7.92815  CP executing      :      1321357703
-----
  
```

```

Inst. buffer fetches/sec :      0.04M      total fetches      :      299961
                               fetch conflicts :      238453
I/O memory refs/sec      :      0.04M      actual refs        :      343362
  avg conflict/ref 0.39:      actual conflicts :      133234
Scalar memory refs/sec   :      1.58M      actual refs        :      12521559
Block memory refs/sec    :      82.92M      actual refs        :      657396235
CPU memory refs/sec      :      84.50M      actual refs        :      669917794
  avg conflict/ref 0.37:      actual conflicts :      248345691
CPU memory writes/sec    :      28.70M      actual refs        :      227572271
CPU memory reads/sec     :      55.79M      actual refs        :      442345523
  
```

```

Group 3: CPU seconds      :      7.91606  CP executing:      1319342956
-----
  
```

```

(octal) type of instruction  inst./CPUsec      actual inst.      % of all inst.
(000-017)jump/special      :      5.06M          40084010          7.64
(020-077)scalar functional unit :  53.35M          422354005          80.48
(100-137)scalar memory      :      1.58M          12521561          2.39
(140-157,175)vector integer/log.:  0.07M           553511            0.11
(160-174)vector floating point :  2.48M          19598137          3.73
(176-177)vector load and store :  3.75M          29689910          5.66
  
```

```

type of operation          ops/CPUsec      actual ops      avg. VL
Vector integer&logical      :      1.13M          8961744          16.19
Vector floating point      :      54.71M          433072172         22.10
Scalar functional unit      :      53.35M          422354005
  
```

B Data from sim

B.1 Statistics for ADM run under sim

Information on instruction buffers:

Total buffer loads = 4788608
CP waiting for loads = 107724483

Total instructions executed = 1414676468

Individual instruction counts:

000= 0 032= 7862614 064=17422846 116= 1088741 150= 101286
001= 0 033= 0 065= 0 117= 3321384 151= 427333
002= 7239876 034= 3258428 066=28906698 120= 4390201 152= 379716
003= 692791 035= 1812299 067=10182043 121=10073745 153= 0
004= 243 036= 1495229 070=10559842 122=12477348 154= 555255
005= 2460711 037= 2243363 071=29801080 123= 7200528 155= 8970
006= 1297636 040=14555441 072= 23062 124= 7550643 156= 922973
007= 2428486 041= 216945 073= 380966 125= 2926717 157= 125
010= 1153637 042=12838353 074=105849772 126= 3222552 160= 461471
011=12517341 043= 3560574 075=78408227 127= 7279491 161= 979748
012= 157493 044= 4952457 076= 3669926 130= 96960 162= 0
013= 1990459 045= 681956 077= 575658 131= 8912260 163= 0
014= 1623772 046= 1675770 100= 289151 132= 5787626 164= 4995356
015= 6721852 047= 4960005 101= 241425 133= 1082988 165=14618020
016= 5070665 050= 2895540 102= 239518 134= 508194 166= 0
017= 3936318 051=15663903 103= 884481 135= 5201997 167= 829771
020=23530526 052= 4450085 104= 1185550 136= 1806498 170= 607854
021= 1888030 053= 3436 105= 1144257 137= 3863376 171=18046941
022=25127781 054= 9854688 106= 1321887 140= 10771 172= 1866907
023=52827819 055= 4537200 107= 9401319 141= 146 173=15173996
024=176524939 056= 290636 110= 99227 142= 425283 174= 968203
025=125279746 057= 426597 111= 628674 143= 35 175= 565245
026= 285474 060=52093114 112= 143461 144= 239 176=35199208
027= 520093 061=31746229 113= 771040 145= 11705 177=20533075
030=200232158 062=12315761 114= 2121604 146= 783987
031=27744055 063=25813379 115= 3427408 147= 276575

Average instruction length = 1.24 parcels

Information on conditional jumps:

Awaiting operand = 22546195 1.6%
Out of buffer = 781372 0.1%
Forward = 1228826 0.1%
Backward = 3266325 0.2%
Not taken = 5238534 0.4%
Total = 9733685 0.7%

Information on unconditional jumps:

Out of buffer = 249213 0.0%
Forward = 303356 0.0%
Backward = 80770 0.0%
Total = 384126 0.0%

Information on return jumps:

Out of buffer = 546864 0.0%
Total = 613838 0.0%

Information on B register jumps:

Out of buffer = 506177 0.0%
Total = 628145 0.0%

Information on various operations:

Scalar floating operations = 28629883
Vector floating operations = 125195107
Vector floating instructions = 17379455
Vector floating average VL = 7
Total floating operations = 153824990

Scalar loads = 20088047

Vector loads = 81674420
Vector load instructions = 10451411
Vector load average VL = 7
Total words loaded = 101762467

Scalar stores = 11594981
Vector stores = 49912360
Vector store instructions = 6094455
Vector store average VL = 8
Total words stored = 61507341

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 97766708 |
| 2 | 17332262 |
| 4 | 4986608 |
| 8 | 0 |
| 16 | 2077736 |
| 32 | 0 |
| 64 | 8101752 |
| 128 | 0 |
| 256 | 0 |

B.2 Statistics for ARC2D run under sim

Information on instruction buffers:

Total buffer loads = 111903
CP waiting for loads = 2841513

Total instructions executed = 400022669

Individual instruction counts:

```
000=      0 032= 9966956 064=  97867 116=   2032 150=   9366
001=      0 033=      0 065=      0 117=  25794 151=  428062
002= 6828711 034=  18948 066=  61496 120=  82317 152=   9164
003=  12388 035=  17233 067=  28122 121=  21984 153=      0
004=   123 036=  12715 070=  29022 122=  11283 154= 1882542
005= 348718 037= 204962 071= 8656153 123=   943 155=   2590
006=  44622 040= 717834 072=   210 124=  31497 156=   2866
007= 337934 041=   4031 073= 999626 125= 125571 157=   2142
010= 344994 042= 2047205 074=21342005 126= 182379 160=  327133
011= 6231678 043=  664831 075=11487542 127= 1331978 161= 2375679
012=  23232 044= 1646450 076=  480656 130=  25886 162=      0
013=  36969 045=   7293 077= 107312 131=   7744 163=      0
014= 102557 046=  41956 100=  28605 132= 10957 164= 5597103
015= 378914 047= 1610942 101=   3984 133=  34202 165=14765850
016=  38665 050=  11881 102=  11630 134=  14013 166=      0
017= 1609235 051= 2444971 103=   9910 135=   5560 167= 1431962
020= 9311200 052=  19349 104=  11900 136=  23891 170=  372138
021=   1203 053=   4731 105=   8570 137= 107059 171= 8465436
022= 1232761 054= 1025246 106=  34584 140= 194099 172=  653332
023=19614159 055= 1308279 107=  83609 141=    16 173= 6141356
024=58682786 056=   33678 110=  15635 142= 359221 174= 2363477
025=49435041 057=  53617 111=   2834 143=   1520 175= 1359860
026=   1581 060=11660969 112=   5986 144=    17 176=26585404
027=   9677 061=  571352 113=   9568 145=    140 177=10921143
030=80061675 062=   78144 114=   5809 146=  325133
031= 439817 063= 197461 115=  11882 147=  366637
```

Average instruction length = 1.08 parcels

Information on conditional jumps:

```
Awaiting operand =  1346478  0.3%
Out of buffer =    30982  0.0%
Forward =    109585  0.0%
Backward =    568074  0.1%
Not taken =    521160  0.1%
Total =  1198819  0.3%
```

Information on unconditional jumps:

```
Out of buffer =    7880  0.0%
Forward =    25983  0.0%
Backward =    4282  0.0%
Total =    30265  0.0%
```

Information on return jumps:

```
Out of buffer =    20480  0.0%
Total =    52945  0.0%
```

Information on B register jumps:

```
Out of buffer =    13807  0.0%
Total =    62373  0.0%
```

Information on various operations:

```
Scalar floating operations =  71306
Vector floating operations = 270337421
Vector floating instructions = 5137981
Vector floating average VL =    52
Total floating operations = 270408727
```

Scalar loads = 384690

Vector loads = 166086687
Vector load instructions = 3220912
Vector load average VL = 51
Total words loaded = 166471377

Scalar stores = 168637
Vector stores = 68471670
Vector store instructions = 1323213
Vector store average VL = 51
Total words stored = 68640307

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 234541613 |
| 2 | 16744 |
| 4 | 0 |
| 8 | 0 |
| 16 | 0 |
| 32 | 0 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

B.3 Statistics for BDNA run under sim

Information on instruction buffers:

Total buffer loads = 4001040
CP waiting for loads = 90735724

Total instructions executed = 300091625

Individual instruction counts:

000= 0 032= 175799 064= 2348880 116= 639 150= 296739
001= 0 033= 0 065= 0 117= 227820 151= 897407
002= 2943777 034= 193894 066= 413847 120= 2034304 152= 217492
003= 349677 035= 193769 067= 352 121= 2880050 153= 0
004= 285 036= 155870 070= 373 122= 501944 154= 2831392
005= 1875038 037= 478403 071= 2959987 123= 106218 155= 97287
006= 1381581 040= 7905515 072= 8 124= 2603408 156= 111892
007= 1265737 041= 642 073= 1619783 125= 308895 157= 35854
010= 1340129 042= 4397803 074= 8544071 126= 3322458 160= 829416
011= 1561979 043= 4657428 075= 8986861 127= 448029 161= 2046182
012= 1679769 044= 7882663 076= 1573663 130= 325148 162= 0
013= 3882061 045= 2925133 077= 209724 131= 244568 163= 0
014= 2295817 046= 1514763 100= 1707762 132= 275257 164= 1712925
015= 3837420 047= 250735 101= 1530 133= 82672 165= 4794912
016= 4245138 050= 171890 102= 606412 134= 2508706 166= 0
017= 1984439 051=12118885 103= 54601 135= 191008 167= 839331
020=14304222 052= 1725046 104= 253980 136= 117204 170= 1718946
021= 111 053= 89449 105= 15575 137= 685146 171= 6472454
022= 6026129 054=11302805 106= 123274 140= 295274 172= 1541562
023= 8422376 055= 5543546 107= 279772 141= 35876 173= 1401345
024=14036737 056= 5422357 110= 822958 142= 555848 174= 1852479
025= 8575301 057= 459915 111= 1999 143= 24 175= 1722329
026= 240750 060=14011023 112= 1468 144= 71555 176= 7803658
027= 585169 061= 4054677 113= 1475 145= 35848 177= 5620712
030=28580950 062= 1534120 114= 73705 146= 784315
031= 8578721 063= 2833211 115= 10263 147= 120

Average instruction length = 1.36 parcels

Information on conditional jumps:

Awaiting operand = 58390428 19.5%
Out of buffer = 582664 0.2%
Forward = 4226457 1.4%
Backward = 3253097 1.1%
Not taken = 13347198 4.4%
Total = 20826752 6.9%

Information on unconditional jumps:

Out of buffer = 465212 0.2%
Forward = 345601 0.1%
Backward = 1035980 0.3%
Total = 1381581 0.5%

Information on return jumps:

Out of buffer = 834436 0.3%
Total = 1265737 0.4%

Information on B register jumps:

Out of buffer = 545499 0.2%
Total = 1875038 0.6%

Information on various operations:

Scalar floating operations = 7130783
Vector floating operations = 1174174207
Vector floating instructions = 23209552
Vector floating average VL = 50
Total floating operations = 1181304990

Scalar loads = 15248212

Vector loads = 374630858
Vector load instructions = 7803658
Vector load average VL = 48
Total words loaded = 389879070

Scalar stores = 5570036
Vector stores = 268746997
Vector store instructions = 5620712
Vector store average VL = 47
Total words stored = 274317033

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 619284599 |
| 2 | 19194 |
| 4 | 18632256 |
| 8 | 2285816 |
| 16 | 0 |
| 32 | 0 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

B.4 Statistics for DYFESM run under sim

Information on instruction buffers:

Total buffer loads = 1309030
CP waiting for loads = 30185739

Total instructions executed = 580255910

Individual instruction counts:

```
000=      0 032= 6256728 064=  94242 116=      60 150=  53927
001=      0 033=      0 065=      6 117= 446592 151= 3747163
002=21780112 034=  730077 066=  56081 120= 370000 152= 3659999
003=      766 035=  371093 067=  27366 121= 2271069 153=      0
004=     139 036=  279425 070=  42390 122= 492954 154=     508
005= 395333 037= 3913867 071=14294999 123= 2236329 155=  53920
006= 264522 040= 677370 072=      8 124=  51667 156= 1537665
007= 388141 041=      59 073=      232 125= 1749901 157=      92
010= 3042437 042= 8523731 074=20376210 126= 111444 160= 768784
011= 9040324 043= 3307542 075=16698645 127= 4511063 161=      9
012=  25546 044= 6224891 076=29279323 130=  27802 162=      0
013= 419000 045=  18863 077= 3746534 131=  6202 163=      0
014= 2255164 046= 499083 100=  54763 132= 12014 164= 2320219
015= 3546022 047= 6728296 101=  1825 133=  15679 165= 3697112
016= 4056257 050= 456893 102=  5727 134= 176432 166=      0
017= 2739183 051= 7455608 103=  1001 135= 2003242 167=      6
020=19897916 052= 3039694 104=  6086 136= 26926 170=      3
021= 1926621 053=      2324 105=      649 137= 3346874 171=20585614
022=48679899 054= 473036 106=  8403 140=  2645 172= 1541572
023=28474932 055= 598633 107= 170460 141=  104 173=  29092
024=29374170 056=  33741 110= 17625 142= 25578 174=      6
025=23871056 057= 14788 111=      55 143=  12 175=  175
026=      3972 060=12563007 112= 44892 144=  187 176=24066637
027=      5590 061= 6841160 113= 40675 145=  124 177= 6430290
030=104162827 062=28682693 114= 446674 146=  46
031= 3428311 063= 2589465 115= 402993 147=  0
```

Average instruction length = 1.19 parcels

Information on conditional jumps:

```
Awaiting operand = 61071581 10.5%
Out of buffer = 101876 0.0%
Forward = 3510248 0.6%
Backward = 5428488 0.9%
Not taken = 16185197 2.8%
Total = 25123933 4.3%
```

Information on unconditional jumps:

```
Out of buffer = 41728 0.0%
Forward = 253868 0.0%
Backward = 10654 0.0%
Total = 264522 0.0%
```

Information on return jumps:

```
Out of buffer = 337051 0.1%
Total = 388141 0.1%
```

Information on B register jumps:

```
Out of buffer = 188131 0.0%
Total = 395333 0.1%
```

Information on various operations:

```
Scalar floating operations = 31492243
Vector floating operations = 520786619
Vector floating instructions = 28942417
Vector floating average VL = 17
Total floating operations = 552278862
```

Scalar loads = 12043341

Vector loads = 461607529
Vector load instructions = 24066637
Vector load average VL = 19
Total words loaded = 473650870

Scalar stores = 7014737
Vector stores = 340139313
Vector store instructions = 6430290
Vector store average VL = 52
Total words stored = 347154050

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 783747378 |
| 2 | 356949 |
| 4 | 9753328 |
| 8 | 101080 |
| 16 | 0 |
| 32 | 1986430 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

B.5 Statistics for FLO52 run under sim

Information on instruction buffers:

Total buffer loads = 623701
CP waiting for loads = 14602791

Total instructions executed = 239155544

Individual instruction counts:

```
000=      0 032= 1054956 064= 1482148 116=      40 150=   8721
001=      0 033=      0 065=      0 117=   30807 151=  63691
002= 5874054 034=   80695 066= 4212443 120=  603494 152=    523
003=   5538 035=   72078 067= 1119270 121=  464578 153=     0
004=    107 036=   53162 070= 1185555 122=  418856 154= 153283
005= 253116 037=  175010 071= 7175647 123=  395640 155=   8415
006=  77549 040= 1767338 072=     20 124=  441374 156=    667
007= 195030 041=     632 073=   83949 125= 1268108 157=    959
010= 508596 042= 2672984 074=15167055 126= 177078 160=  32262
011= 5483487 043=  409964 075= 8767433 127= 173079 161=  667137
012= 228788 044= 2617748 076=  11434 130= 119599 162=     0
013= 212406 045= 230737 077=  33156 131=  66499 163=     0
014= 251587 046=   92554 100=  437226 132=  39774 164= 2269324
015= 576819 047= 2215335 101=    3621 133=  70520 165= 3290161
016= 199610 050=  420681 102=  19126 134= 110896 166=     0
017= 2582617 051= 2578308 103=   3577 135= 113358 167=  586001
020=15142611 052=  455117 104=   4926 136= 130887 170=  69664
021= 287692 053=     1941 105=     588 137= 347012 171= 2723116
022= 1353652 054=  624724 106=  29924 140=  66527 172=   5536
023=14112710 055= 1074827 107=  36244 141=     64 173= 5284239
024=19460105 056=  235635 110= 108841 142=  36691 174=  660200
025=18286526 057=   78737 111=   6495 143=    911 175= 249175
026=  22346 060= 5941888 112=   4122 144=    114 176=12438707
027=  21574 061= 1800912 113=   3355 145=    988 177= 4876587
030=45050737 062= 2573839 114=   5754 146=   75348
031= 1490079 063= 1640220 115=  41965 147= 123702
```

Average instruction length = 1.23 parcels

Information on conditional jumps:

```
Awaiting operand = 18251706 7.6%
Out of buffer = 101351 0.0%
Forward = 677726 0.3%
Backward = 3391539 1.4%
Not taken = 5974645 2.5%
Total = 10043910 4.2%
```

Information on unconditional jumps:

```
Out of buffer = 29246 0.0%
Forward = 49560 0.0%
Backward = 27989 0.0%
Total = 77549 0.0%
```

Information on return jumps:

```
Out of buffer = 135069 0.1%
Total = 195030 0.1%
```

Information on B register jumps:

```
Out of buffer = 62709 0.0%
Total = 253116 0.1%
```

Information on various operations:

```
Scalar floating operations = 12213475
Vector floating operations = 629242369
Vector floating instructions = 15587640
Vector floating average VL = 40
Total floating operations = 641455844
```

Scalar loads = 4477439

Vector loads = 513976589
Vector load instructions = 12438707
Vector load average VL = 41
Total words loaded = 518454028

Scalar stores = 1199924
Vector stores = 195138142
Vector store instructions = 4876587
Vector store average VL = 40
Total words stored = 196338066

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 648636249 |
| 2 | 60462850 |
| 4 | 0 |
| 8 | 1632 |
| 16 | 0 |
| 32 | 0 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

B.6 Statistics for MDG run under sim

Information on instruction buffers:

Total buffer loads = 5132304
CP waiting for loads = 120113328

Total instructions executed = 8510916422

Individual instruction counts:

000= 0 032= 2163018 064=200124876 116= 47 150=20920096
001= 0 033= 0 065= 0 117= 5428 151=25810244
002=104110914 034= 46530 066=234160406 120=161542713 152= 1853875
003=21046757 035= 43229 067=69736480 121=110465386 153= 0
004= 105 036= 7655 070=160793788 122=46918877 154=31450200
005=46431679 037=23087971 071=59485252 123=15318491 155= 7354068
006= 18284 040=80827698 072= 8 124= 7488701 156= 7354069
007=46426426 041= 29 073=65016963 125=18921533 157= 621
010=12280369 042=103879451 074=739237440 126=57529778 160=39832936
011=24124243 043=80943129 075=345642095 127=145005777 161=39217598
012= 26700 044=22651688 076=44897081 130= 13776 162= 0
013= 78561 045=11418728 077=14242469 131=64021113 163= 0
014=69293477 046= 6653940 100= 30911 132=11321526 164=15023092
015=16075211 047=28574254 101=48297102 133= 105983 165=82943594
016=18991000 050= 6578 102= 7490173 134= 3728724 166= 0
017=59079341 051=168229929 103= 670 135=43286548 167=15320189
020=272728211 052= 7507113 104= 1309 136=269443910 170=49842404
021= 3726530 053= 3364 105= 120 137=38051392 171=54953427
022=168910346 054=120764418 106= 3867289 140=25257664 172=33764971
023=71750742 055=116367824 107=26090383 141= 629 173=30230341
024=768895207 056= 80915 110= 12391 142= 5515164 174=24509549
025=382944691 057=26419791 111= 30 143= 8 175=71229047
026=10031875 060=156982360 112= 155 144= 1237 176=116015421
027= 12153 061=102804996 113= 1112 145= 757 177=63962666
030=523421209 062=456994129 114= 1108 146=31589768
031=46518991 063=521279758 115= 1893 147= 63

Average instruction length = 1.37 parcels

Information on conditional jumps:

Awaiting operand = 41225957 0.5%
Out of buffer = 916655 0.0%
Forward = 2780898 0.0%
Backward = 1159689 0.0%
Not taken = 8077151 0.1%
Total = 12017738 0.1%

Information on unconditional jumps:

Out of buffer = 136 0.0%
Forward = 544 0.0%
Backward = 235 0.0%
Total = 779 0.0%

Information on return jumps:

Out of buffer = 697876 0.0%
Total = 2811665 0.0%

Information on B register jumps:

Out of buffer = 246 0.0%
Total = 2811743 0.0%

Information on various operations:

Scalar floating operations = 99537047
Vector floating operations = 133447071
Vector floating instructions = 22653442
Vector floating average VL = 5
Total floating operations = 232984118

Scalar loads = 39383463

Vector loads = 64865098
Vector load instructions = 6827821
Vector load average VL = 9
Total words loaded = 104248561

Scalar stores = 25609520
Vector stores = 38208457
Vector store instructions = 3790980
Vector store average VL = 10
Total words stored = 63817977

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 94102029 |
| 2 | 644 |
| 4 | 0 |
| 8 | 0 |
| 16 | 0 |
| 32 | 0 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

B.7 Statistics for MG3D run under sim

Information on instruction buffers:

Total buffer loads = 3345135
CP waiting for loads = 85450902

Total instructions executed = 6056939408

Individual instruction counts:

000= 0 032=54902957 064=38167093 116= 3593886 150= 135
001= 0 033= 0 065= 0 117= 2641061 151= 298859
002=39717530 034= 2552771 066= 598667 120= 4520829 152= 1136
003= 216831 035= 2112482 067= 8918648 121= 372437 153= 0
004= 6984 036= 2100888 070= 8918651 122= 807014 154= 1136
005= 3815794 037= 3851381 071=138166914 123= 2774465 155= 117
006= 1753762 040= 5874854 072= 8 124= 5787658 156= 528
007= 2142980 041= 13261 073= 102012 125=15045457 157= 1120
010= 935008 042=36429953 074=444977640 126=39866303 160= 0
011=58149931 043= 6247843 075=267319514 127=80936270 161= 1323053
012= 64440 044=31377989 076= 1239 130= 15932 162= 0
013= 2248101 045= 74221 077= 291078 131= 224653 163= 0
014= 1037323 046= 6228373 100= 78613 132= 113817 164=152657080
015= 8444897 047=25164754 101= 3913 133= 49880 165=11239976
016= 9939523 050=16943234 102= 1233881 134= 1208181 166= 0
017=21720115 051=15475365 103= 5110445 135= 9729510 167= 1323050
020=127368977 052= 1435109 104= 1393064 136=20917844 170= 95461
021= 11754 053= 1870 105=16299278 137=84292640 171=122915308
022=28700867 054= 6918516 106=11793022 140= 9043 172= 8653398
023=303278074 055=13933249 107=70890127 141= 1124 173=119119512
024=988637359 056= 258647 110= 39131 142= 97191 174= 1323050
025=656256159 057= 135498 111= 44897 143= 16 175= 97036
026= 1796 060=211996662 112= 157174 144= 3330 176=196447494
027= 404862 061=169098285 113= 450572 145= 2252 177=155390346
030=979762036 062= 8723251 114=13295604 146= 94925
031=30831770 063=94074729 115= 3225805 147= 95890

Average instruction length = 1.19 parcels

Information on conditional jumps:

Awaiting operand = 12994430 0.2%
Out of buffer = 549568 0.0%
Forward = 717570 0.0%
Backward = 3137198 0.1%
Not taken = 5556534 0.1%
Total = 9411302 0.2%

Information on unconditional jumps:

Out of buffer = 145874 0.0%
Forward = 149881 0.0%
Backward = 866 0.0%
Total = 150747 0.0%

Information on return jumps:

Out of buffer = 151673 0.0%
Total = 187366 0.0%

Information on B register jumps:

Out of buffer = 186057 0.0%
Total = 332779 0.0%

Information on various operations:

Scalar floating operations = 14298643
Vector floating operations = 996941158
Vector floating instructions = 39054864
Vector floating average VL = 25
Total floating operations = 1011239801

Scalar loads = 23101254

Vector loads = 586638585
Vector load instructions = 18624597
Vector load average VL = 31
Total words loaded = 609739839

Scalar stores = 12362853
Vector stores = 420572033
Vector store instructions = 14589978
Vector store average VL = 28
Total words stored = 432934886

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 742816660 |
| 2 | 264393958 |
| 4 | 0 |
| 8 | 0 |
| 16 | 0 |
| 32 | 0 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

B.8 Statistics for OCEAN run under sim

Information on instruction buffers:

Total buffer loads = 250479
CP waiting for loads = 5597637

Total instructions executed = 959509361

Individual instruction counts:

```
000=      0 032= 4361351 064= 5512219 116=      32 150=    2373
001=      0 033=      0 065=      0 117=   69873 151=   48432
002=11462997 034=  308418 066=35750502 120=  710839 152=  96016
003=  49841 035=  155479 067=  448040 121=32823020 153=      0
004=      79 036=  155594 070=  448061 122=  384378 154=  48901
005= 165160 037=  460709 071=14890424 123=  96142 155=   1972
006= 378225 040=  477726 072=      78 124=  97052 156=    803
007= 160352 041=      23 073=    901 125= 1881882 157=      4
010= 150708 042= 6082951 074=59187301 126=  882355 160=    788
011=25958639 043=  432085 075=36489602 127=33244045 161= 321714
012=  12799 044= 3004977 076= 3476546 130=  196033 162=      0
013= 169378 045=   5487 077=  48075 131=32438563 163=      0
014= 399393 046=  412923 100=  57079 132=32438959 164= 6942859
015= 200148 047= 3141496 101=    753 133=  70570 165= 4239324
016= 1885153 050= 1376593 102=  4036 134=  38177 166=      0
017= 4601778 051= 4388214 103=    532 135= 1355622 167= 321668
020=22053226 052=  156892 104=    908 136=  618337 170=  423626
021= 2822857 053=    480 105=    106 137=33706532 171= 9505468
022= 6893666 054=  273256 106=   1748 140=   4152 172= 1780934
023=44265413 055= 2879292 107=  3047 141=      7 173= 7498077
024=49073517 056=  23116 110=   9068 142=  1451 174= 321668
025=27348980 057=  4050 111=    23 143=    10 175=   2294
026=  1292 060=24052213 112=    641 144=    389 176=23978528
027=    530 061=24680521 113=   1774 145=    822 177=20421926
030=176300252 062=34506981 114=  14104 146=  49529
031= 4708284 063=65321638 115=  370679 147=   1836
```

Average instruction length = 1.45 parcels

Information on conditional jumps:

```
Awaiting operand =  4386605  0.5%
Out of buffer =    72174  0.0%
Forward =    152675  0.0%
Backward =   3759453  0.4%
Not taken =   1643185  0.2%
Total =   5555313  0.6%
```

Information on unconditional jumps:

```
Out of buffer =    29886  0.0%
Forward =    62165  0.0%
Backward =    1036  0.0%
Total =    63201  0.0%
```

Information on return jumps:

```
Out of buffer =    22350  0.0%
Total =    27097  0.0%
```

Information on B register jumps:

```
Out of buffer =    3267  0.0%
Total =    27910  0.0%
```

Information on various operations:

```
Scalar floating operations = 23589909
Vector floating operations = 230957236
Vector floating instructions = 5205967
Vector floating average VL = 44
Total floating operations = 254547145
```

Scalar loads = 11663176

Vector loads = 188198841
Vector load instructions = 3985903
Vector load average VL = 47
Total words loaded = 199862017

Scalar stores = 16837397
Vector stores = 172264623
Vector store instructions = 3395304
Vector store average VL = 50
Total words stored = 189102020

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 134011377 |
| 2 | 212980911 |
| 4 | 13471176 |
| 8 | 0 |
| 16 | 0 |
| 32 | 0 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

B.9 Statistics for QCD run under sim

Information on instruction buffers:

Total buffer loads = 5789824
CP waiting for loads = 128938208

Total instructions executed = 1234308873

Individual instruction counts:

```
000=      0 032=      653 064=14735164 116=      38 150=      219
001=      0 033=      0 065=      1 117=     1279 151=    57354
002= 3021775 034= 3620013 066=109001827 120=48655924 152=      10
003=      2250 035= 1810658 067= 2156310 121=18519591 153=      0
004=      108 036= 1810309 070= 3041047 122= 1869179 154=     250
005= 2551718 037= 2338744 071=11532877 123= 4346574 155=     169
006= 2310958 040=32359798 072=      28 124= 4397295 156=     592
007= 2551498 041=      32807 073=      581 125= 6849764 157=      49
010= 386821 042= 8572381 074=233228926 126= 76110 160=      0
011=13825992 043= 5865078 075=120852150 127= 9815181 161=     13
012=      1154 044=10153229 076=      565 130=14762273 162=      0
013=      4553 045= 445637 077= 57530 131=10806125 163=      0
014=12914670 046= 3627847 100= 152806 132= 9069649 164=     97
015= 4064657 047= 15295 101=      483 133= 504 165=      0
016= 4132711 050= 8488479 102=      353 134=      75 166=      0
017= 7190551 051= 9301752 103=      212 135= 344590 167=      0
020=10526413 052= 174471 104=      365 136= 66580 170=     13
021= 4984868 053=      1220 105=      140 137= 3048320 171=   196656
022=28917123 054= 5445136 106=     1147 140=     188 172=    194
023=21324001 055=13336601 107= 34126 141=      7 173=      0
024=47588072 056= 4257 110= 1784 142=     65 174=      0
025=42390543 057= 2121 111=      24 143=      6 175=     683
026=      222 060=38945349 112=      39 144=      21 176= 2176030
027=      671 061=60838517 113=      77 145=      45 177= 2037714
030=61118501 062=63792073 114=      20 146=    1835
031= 3067526 063=58548940 115=      213 147=      96
```

Average instruction length = 1.33 parcels

Information on conditional jumps:

Awaiting operand = 174725916 14.2%

Out of buffer = 2659990 0.2%

Forward = 12630670 1.0%

Backward = 8818378 0.7%

Not taken = 8907245 0.7%

Total = 30356293 2.5%

Information on unconditional jumps:

Out of buffer = 880576 0.1%

Forward = 213103 0.0%

Backward = 917930 0.1%

Total = 1131033 0.1%

Information on return jumps:

Out of buffer = 91583 0.0%

Total = 244982 0.0%

Information on B register jumps:

Out of buffer = 244625 0.0%

Total = 245063 0.0%

Information on various operations:

Scalar floating operations = 73526841

Vector floating operations = 338

Vector floating instructions = 248

Vector floating average VL = 1

Total floating operations = 73527179

Scalar loads = 45552261

Vector loads = 20330477
Vector load instructions = 704962
Vector load average VL = 28
Total words loaded = 65882738

Scalar stores = 13324224
Vector stores = 20994453
Vector store instructions = 730471
Vector store average VL = 28
Total words stored = 34318677

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 41323750 |
| 2 | 1036 |
| 4 | 0 |
| 8 | 8 |
| 16 | 0 |
| 32 | 0 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

B.10 Statistics for SPEC77 run under sim

Information on instruction buffers:

Total buffer loads = 4741071
CP waiting for loads = 107016019

Total instructions executed = 2897930523

Individual instruction counts:

000= 0 032= 5416207 064=14194389 116= 26974 150= 474076
001= 0 033= 0 065= 0 117= 1401948 151= 2260041
002=63350425 034= 1996349 066= 6542451 120= 6347007 152=16604848
003= 466563 035= 1553751 067= 382699 121=15414969 153= 0
004= 519 036= 1179573 070= 383599 122= 5040088 154= 1440294
005= 5269946 037= 2001479 071=30223964 123= 5163262 155= 223379
006= 4235549 040=41858039 072= 8 124= 6513774 156= 733489
007= 3990176 041= 4326577 073= 824377 125= 3670159 157= 95812
010= 3670165 042=29372422 074=164396920 126= 7689587 160= 383351
011=26903662 043=15954851 075=121679616 127= 9850716 161= 516301
012= 6817314 044=37160125 076=127812022 130= 968460 162= 0
013=16264671 045=14721287 077= 1207891 131= 722397 163= 0
014= 9677707 046= 6262662 100= 8278179 132= 663157 164=38217969
015=13874784 047= 5249934 101= 46030 133= 498172 165=21301480
016=17426724 050= 1362041 102= 512881 134= 2447607 166= 0
017=14492348 051=51440149 103= 107820 135= 5434411 167= 352040
020=54838691 052=11088104 104= 321333 136= 5294992 170= 657488
021=30406466 053= 594485 105= 25344 137=14810103 171=62800195
022=145308551 054=43975389 106= 474144 140= 1524582 172= 1061130
023=111661306 055=30464125 107= 2404415 141= 92465 173=25450715
024=255922923 056=14759094 110= 2704409 142= 593383 174= 374705
025=214343750 057= 3945364 111= 56 143= 3383 175= 1073524
026= 1730619 060=151962511 112= 9405 144= 184643 176=92428891
027= 2920613 061=21182864 113= 45107 145= 92297 177=39746022
030=326281240 062=125518813 114= 173042 146= 489454
031=48162409 063=11818551 115= 834299 147= 30516

Average instruction length = 1.21 parcels

Information on conditional jumps:

Awaiting operand = 54509113 1.9%
Out of buffer = 1452827 0.1%
Forward = 4888146 0.2%
Backward = 4326125 0.1%
Not taken = 10949513 0.4%
Total = 20163784 0.7%

Information on unconditional jumps:

Out of buffer = 258222 0.0%
Forward = 654429 0.0%
Backward = 356798 0.0%
Total = 1011227 0.0%

Information on return jumps:

Out of buffer = 874194 0.0%
Total = 1078622 0.0%

Information on B register jumps:

Out of buffer = 496923 0.0%
Total = 1380047 0.0%

Information on various operations:

Scalar floating operations = 24857600
Vector floating operations = 300798914
Vector floating instructions = 24827421
Vector floating average VL = 12
Total floating operations = 325656514

Scalar loads = 13380197

Vector loads = 192014892
Vector load instructions = 15229270
Vector load average VL = 12
Total words loaded = 205395089

Scalar stores = 8117873
Vector stores = 74604217
Vector store instructions = 6324447
Vector store average VL = 11
Total words stored = 82722090

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 45005144 |
| 2 | 55926309 |
| 4 | 92071720 |
| 8 | 0 |
| 16 | 0 |
| 32 | 71812560 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

B.11 Statistics for SPICE run under sim

Information on instruction buffers:

Total buffer loads = 6088258
CP waiting for loads = 140120388

Total instructions executed = 327496357

Individual instruction counts:

| | | | | | | | | | |
|------|----------|------|----------|------|----------|------|----------|------|---------|
| 000= | 0 | 032= | 1223 | 064= | 6424965 | 116= | 79966 | 150= | 1478 |
| 001= | 0 | 033= | 0 | 065= | 3 | 117= | 10035 | 151= | 2843 |
| 002= | 2052029 | 034= | 420548 | 066= | 13680497 | 120= | 23559370 | 152= | 208 |
| 003= | 757 | 035= | 256402 | 067= | 2395655 | 121= | 4029322 | 153= | 0 |
| 004= | 106 | 036= | 233660 | 070= | 2923715 | 122= | 6221193 | 154= | 1770 |
| 005= | 1334702 | 037= | 233605 | 071= | 4156615 | 123= | 3484491 | 155= | 1485 |
| 006= | 4736676 | 040= | 3535721 | 072= | 7552 | 124= | 3417428 | 156= | 9324 |
| 007= | 856388 | 041= | 50010 | 073= | 13637 | 125= | 825243 | 157= | 188 |
| 010= | 375314 | 042= | 4680808 | 074= | 22590107 | 126= | 3037265 | 160= | 0 |
| 011= | 1700597 | 043= | 1991225 | 075= | 18528253 | 127= | 10617308 | 161= | 378879 |
| 012= | 36639 | 044= | 1289171 | 076= | 41951 | 130= | 13681330 | 162= | 0 |
| 013= | 63457 | 045= | 2560072 | 077= | 5238 | 131= | 2325762 | 163= | 0 |
| 014= | 6249173 | 046= | 6367469 | 100= | 6419867 | 132= | 746674 | 164= | 0 |
| 015= | 3130673 | 047= | 922278 | 101= | 1529 | 133= | 263555 | 165= | 375102 |
| 016= | 3208033 | 050= | 1926231 | 102= | 12567 | 134= | 401970 | 166= | 0 |
| 017= | 2910245 | 051= | 5377487 | 103= | 1139 | 135= | 413532 | 167= | 375102 |
| 020= | 6264794 | 052= | 387827 | 104= | 3388 | 136= | 336790 | 170= | 3775 |
| 021= | 822 | 053= | 721 | 105= | 579 | 137= | 4085101 | 171= | 246246 |
| 022= | 2214046 | 054= | 1338557 | 106= | 137810 | 140= | 3076 | 172= | 6 |
| 023= | 14299914 | 055= | 3313638 | 107= | 1035926 | 141= | 194 | 173= | 375102 |
| 024= | 6838637 | 056= | 122983 | 110= | 29481 | 142= | 4009 | 174= | 375102 |
| 025= | 7974757 | 057= | 61542 | 111= | 24 | 143= | 6 | 175= | 13585 |
| 026= | 52731 | 060= | 4854121 | 112= | 1227 | 144= | 387 | 176= | 1787037 |
| 027= | 32650 | 061= | 9157491 | 113= | 1768 | 145= | 3989 | 177= | 778749 |
| 030= | 24245375 | 062= | 11917481 | 114= | 739 | 146= | 151 | | |
| 031= | 1160005 | 063= | 16030215 | 115= | 38873 | 147= | 118 | | |

Average instruction length = 1.65 parcels

Information on conditional jumps:

Awaiting operand = 80017895 24.4%
Out of buffer = 987007 0.3%
Forward = 5399798 1.6%
Backward = 1593775 0.5%
Not taken = 10680558 3.3%
Total = 17674131 5.4%

Information on unconditional jumps:

Out of buffer = 1607031 0.5%
Forward = 2197529 0.7%
Backward = 2539147 0.8%
Total = 4736676 1.4%

Information on return jumps:

Out of buffer = 599752 0.2%
Total = 856388 0.3%

Information on B register jumps:

Out of buffer = 329190 0.1%
Total = 1334702 0.4%

Information on various operations:

Scalar floating operations = 53372531
Vector floating operations = 4133286
Vector floating instructions = 2129314
Vector floating average VL = 1
Total floating operations = 57505817

Scalar loads = 62804425

Vector loads = 4151372
Vector load instructions = 1787037
Vector load average VL = 2
Total words loaded = 66955797

Scalar stores = 22416827
Vector stores = 2653833
Vector store instructions = 778749
Vector store average VL = 3
Total words stored = 25070660

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 6743428 |
| 2 | 57092 |
| 4 | 0 |
| 8 | 4640 |
| 16 | 0 |
| 32 | 0 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

B.12 Statistics for TRACK run under sim

Information on instruction buffers:

Total buffer loads = 3934782
CP waiting for loads = 88821048

Total instructions executed = 497090002

Individual instruction counts:

| | | | | | | | | | |
|------|----------|------|----------|------|----------|------|----------|------|---------|
| 000= | 0 | 032= | 1959468 | 064= | 3120616 | 116= | 45 | 150= | 2351 |
| 001= | 0 | 033= | 0 | 065= | 1 | 117= | 223181 | 151= | 98718 |
| 002= | 1222059 | 034= | 1000803 | 066= | 35137603 | 120= | 21828945 | 152= | 3512 |
| 003= | 4012 | 035= | 530737 | 067= | 1165864 | 121= | 4032445 | 153= | 0 |
| 004= | 128 | 036= | 1515290 | 070= | 1360244 | 122= | 12988109 | 154= | 2894 |
| 005= | 1203417 | 037= | 555470 | 071= | 5393001 | 123= | 1617493 | 155= | 1348 |
| 006= | 283208 | 040= | 2327348 | 072= | 932 | 124= | 300220 | 156= | 27027 |
| 007= | 1186256 | 041= | 26 | 073= | 39044 | 125= | 343322 | 157= | 11062 |
| 010= | 176146 | 042= | 14881514 | 074= | 77005667 | 126= | 145730 | 160= | 0 |
| 011= | 15655532 | 043= | 977835 | 075= | 34735115 | 127= | 5457025 | 161= | 465 |
| 012= | 84074 | 044= | 981690 | 076= | 424512 | 130= | 15085440 | 162= | 0 |
| 013= | 243070 | 045= | 1053612 | 077= | 75677 | 131= | 269712 | 163= | 0 |
| 014= | 810639 | 046= | 15978182 | 100= | 172084 | 132= | 148358 | 164= | 520414 |
| 015= | 14450960 | 047= | 588969 | 101= | 5011 | 133= | 314774 | 165= | 250068 |
| 016= | 2658745 | 050= | 1067746 | 102= | 53240 | 134= | 153107 | 166= | 0 |
| 017= | 6479452 | 051= | 4263495 | 103= | 14310 | 135= | 716536 | 167= | 0 |
| 020= | 6059720 | 052= | 150727 | 104= | 24931 | 136= | 1489119 | 170= | 1005 |
| 021= | 63130 | 053= | 11556 | 105= | 154 | 137= | 1762010 | 171= | 454197 |
| 022= | 8455059 | 054= | 3744255 | 106= | 211012 | 140= | 27443 | 172= | 826 |
| 023= | 17615652 | 055= | 1023861 | 107= | 147123 | 141= | 2009 | 173= | 103332 |
| 024= | 20739414 | 056= | 338427 | 110= | 81474 | 142= | 2514 | 174= | 0 |
| 025= | 18495541 | 057= | 66176 | 111= | 32 | 143= | 10 | 175= | 39940 |
| 026= | 43540 | 060= | 15441782 | 112= | 7201 | 144= | 3990 | 176= | 1530124 |
| 027= | 57227 | 061= | 2465528 | 113= | 861 | 145= | 2482 | 177= | 973518 |
| 030= | 41266042 | 062= | 20458028 | 114= | 2616 | 146= | 3441 | | |
| 031= | 1562936 | 063= | 16774731 | 115= | 141 | 147= | 60 | | |

Average instruction length = 1.39 parcels

Information on conditional jumps:

Awaiting operand = 137436260 27.6%
Out of buffer = 561461 0.1%
Forward = 18413551 3.7%
Backward = 15437309 3.1%
Not taken = 6707758 1.3%
Total = 40558618 8.2%

Information on unconditional jumps:

Out of buffer = 51553 0.0%
Forward = 100226 0.0%
Backward = 182982 0.0%
Total = 283208 0.1%

Information on return jumps:

Out of buffer = 704932 0.1%
Total = 1186256 0.2%

Information on B register jumps:

Out of buffer = 523130 0.1%
Total = 1203417 0.2%

Information on various operations:

Scalar floating operations = 78017087
Vector floating operations = 6714448
Vector floating instructions = 1330307
Vector floating average VL = 5
Total floating operations = 84731535

Scalar loads = 47341154

Vector loads = 10155081
Vector load instructions = 1530124
Vector load average VL = 6
Total words loaded = 57496235

Scalar stores = 20254607
Vector stores = 7522699
Vector store instructions = 973518
Vector store average VL = 7
Total words stored = 27777306

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 17350928 |
| 2 | 138362 |
| 4 | 8416 |
| 8 | 131304 |
| 16 | 0 |
| 32 | 0 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

B.13 Statistics for TRFD run under sim

Information on instruction buffers:

Total buffer loads = 299322
CP waiting for loads = 6586989

Total instructions executed = 524803642

Individual instruction counts:

| | | | | | | | | | |
|------|-----------|------|----------|------|----------|------|---------|------|----------|
| 000= | 0 | 032= | 9175308 | 064= | 116389 | 116= | 36 | 150= | 54147 |
| 001= | 0 | 033= | 0 | 065= | 2 | 117= | 773 | 151= | 14 |
| 002= | 10187341 | 034= | 726 | 066= | 56636 | 120= | 955 | 152= | 4 |
| 003= | 275534 | 035= | 703 | 067= | 2711 | 121= | 2098 | 153= | 0 |
| 004= | 80 | 036= | 517 | 070= | 2715 | 122= | 1151142 | 154= | 11 |
| 005= | 1263 | 037= | 2959945 | 071= | 20454625 | 123= | 44224 | 155= | 72 |
| 006= | 1810 | 040= | 286680 | 072= | 36 | 124= | 157235 | 156= | 113292 |
| 007= | 1085 | 041= | 20 | 073= | 56608 | 125= | 3077636 | 157= | 0 |
| 010= | 371995 | 042= | 10634431 | 074= | 58618136 | 126= | 851557 | 160= | 0 |
| 011= | 14076512 | 043= | 59107 | 075= | 32239283 | 127= | 3973919 | 161= | 56724 |
| 012= | 1255 | 044= | 7095015 | 076= | 56720 | 130= | 806 | 162= | 0 |
| 013= | 2038 | 045= | 351 | 077= | 54104 | 131= | 1455624 | 163= | 0 |
| 014= | 6784012 | 046= | 6784915 | 100= | 1320 | 132= | 478404 | 164= | 9685550 |
| 015= | 815255 | 047= | 7262420 | 101= | 355 | 133= | 237553 | 165= | 0 |
| 016= | 569839 | 050= | 113735 | 102= | 167 | 134= | 209270 | 166= | 0 |
| 017= | 6996447 | 051= | 2912831 | 103= | 46 | 135= | 507668 | 167= | 56700 |
| 020= | 14454552 | 052= | 756575 | 104= | 67 | 136= | 120855 | 170= | 56717 |
| 021= | 1 | 053= | 359 | 105= | 125 | 137= | 191447 | 171= | 9628850 |
| 022= | 3539004 | 054= | 116146 | 106= | 319 | 140= | 80 | 172= | 56896 |
| 023= | 53274345 | 055= | 61758 | 107= | 57400 | 141= | 4 | 173= | 0 |
| 024= | 32271360 | 056= | 6354 | 110= | 821 | 142= | 39 | 174= | 56700 |
| 025= | 24052737 | 057= | 57784 | 111= | 20 | 143= | 4 | 175= | 110642 |
| 026= | 54171 | 060= | 17900997 | 112= | 15 | 144= | 0 | 176= | 19538415 |
| 027= | 2801 | 061= | 2133488 | 113= | 53 | 145= | 25 | 177= | 10151501 |
| 030= | 110108754 | 062= | 56805 | 114= | 155 | 146= | 275170 | | |
| 031= | 4163903 | 063= | 398877 | 115= | 107 | 147= | 7 | | |

Average instruction length = 1.16 parcels

Information on conditional jumps:

Awaiting operand = 47328200 9.0%
Out of buffer = 98417 0.0%
Forward = 467189 0.1%
Backward = 7306107 1.4%
Not taken = 21844057 4.2%
Total = 29617353 5.6%

Information on unconditional jumps:

Out of buffer = 279 0.0%
Forward = 930 0.0%
Backward = 880 0.0%
Total = 1810 0.0%

Information on return jumps:

Out of buffer = 819 0.0%
Total = 1085 0.0%

Information on B register jumps:

Out of buffer = 583 0.0%
Total = 1263 0.0%

Information on various operations:

Scalar floating operations = 634135
Vector floating operations = 433070040
Vector floating instructions = 19598137
Vector floating average VL = 22
Total floating operations = 433704175

Scalar loads = 9318565

Vector loads = 433020835
Vector load instructions = 19538415
Vector load average VL = 22
Total words loaded = 442339400

Scalar stores = 3203607
Vector stores = 224361823
Vector store instructions = 10151501
Vector store average VL = 22
Total words stored = 227565430

Vector stride distribution:

| Stride | Words moved |
|--------|-------------|
| 1 | 513103242 |
| 2 | 27455766 |
| 4 | 3735900 |
| 8 | 112979600 |
| 16 | 0 |
| 32 | 0 |
| 64 | 0 |
| 128 | 0 |
| 256 | 0 |

References

- [1] D. H. Bailey, "Vector computer memory bank contention," *IEEE Trans. on Computers*, vol. C-36, (1987), pp. 293–298.
- [2] G. Cybenko, "Supercomputer performance trends and the Perfect Benchmarks," *Supercomputing Review*, April, (1991), pp. 53–60.
- [3] J. L. Larson, "Collecting and interpreting hpm performance data on the Cray Y-MP," *NCSA Datalink*, November-December, (1991).
- [4] A. D. Malony, J. L. Larson, and D. A. Reed, "Tracing application program execution on the Cray X-MP and Cray 2," *Proc. Supercomputing '90*, (1990), pp. 60–73.
- [5] K. A. Robbins, and S. Robbins, "Bus conflicts for logical memory banks on a Cray Y-MP type processor system," *1991 Intl. Conf. on Parallel Processing*, (1991), pp. 21–24.
- [6] K. A. Robbins, and S. Robbins, "Dynamic behavior of memory reference streams for the Perfect Club Benchmarks," *1992 Intl. Conf. on Parallel Processing*, (1992), to appear.
- [7] Vajapeyam, S., Sohi, G. S. and Hsu, W.-C, "An empirical study of the Cray Y-MP processor using the Perfect Club Benchmarks," *18th Intl. Sym. Computer Architecture*, (1991), pp. 170–179.