PDP-K Technical Memorandum #8.10

Title: Programming Benchmarkes: PM 18-3 Problem Set Coded on Small and Medium Processors

Author: Robert Gray

Distribution Keys:

Index Keys: Programs Instruction Set

Revision None

Obsolete: None

Date: March 25, 1970

Abstract

The set of five problems proposed in Interoffice Memo PM 18-3 (PDP-Z Distribution 5/31/68) are coded on the 18 and 36 bit versions of PDP-K. The results are compared with PDP-15, PDP-10, PDP-11, Sigma 3/3, Sigma 5/7, and System 36.

The coding shows that the K performs very well on this set of problems in terms of doing the tasks with few instructions and few memory references.

Though the problems are only a very limited test of the PDP-K's capability, they show the usefulness of several of the K's proposed instructions:

1. COMPARE WITH LIMITS

2. REPEAT

3. CHANGE COUNTER, TEST AND JUMP (AOJ, SOJ, AOS, SOS)

The first two greatly reduced memory references in two of the problems and the third helps generate efficient small loops.

Introduction

This is the first in a series of memos of coding benchmarks comparing the PDP-K with other computers.

This memo, in itself, is not intended to be a complete examination of the various instruction sets. It looks at the intrinsic value of some of the capability of the various machines.

It is our intent to produce "good" code for each of the machines evaluated. To that end, people with experience on particular machines have been sought when possible, to code or checkout our code for efficiency. However, as these have not been actually run, it is inevitable that the careful reader will find errors in the coding or tabulation. Hopefully they will be minor and not effect the all over implications found. We would appreciate such errors being called to our attention. If it is found that significant errors have occurred, it is our intention to revise and reissue this memo.

The PDP-9/15, PDP-11, PDP-10, and PDP-K coding has been checked or coded by "experts." No expert could be found for Sigma 2/3, and since the XDS documentation is sketchy and the instruction set incomplete, it is probable that the coding for this machine (and hence its performance) could be slightly improved relative to the others. System 86 is too new for any experienced people to be available.

Future Parts to this Memo

It is our intention to "run K" on the "Comparative Benchmarks" being set up by Clarke Wegner in the PDP-10 group when they are specified. In addition to the machines included in this part, that will include GE635, 7094, and two variations of the 360 Series.

It is also our intention to code the 635, 1108, and 360/44 on this set of problems as Memo # 1.20.

From time to time other parts will be issued to examine other aspects of the PDP-K's architecture and instruction set. Suggestions and comments on this part, and for future parts, are solicited.

Purpose of this Memo

The purpose of this memo is to compare some aspects of the instruction sets of a number of operating computers with the proposed K instruction set. The method used was to code the set of five problems set forth by John Cohen and Larry McGowan in PM18-3 (May 31, 1968). This set of problems had been coded for a number of mini computers and was aimed at a set of typical problems for mini computers.

Using this set of problems as sole criteria to evaluate a set of instructions would be foolish. At the K level of computer it totally neglects the floating point capability, most EAĖ capability, and fails to examine subroutine calls for ability to be re-entrant and recursive.

As a tool for valuation, this memo was thought to be useful on two counts. First the results are already known for this set of problems on most 16-bit machines (or soon will be) and hence we felt it would be instructive to see how much improvement would be obtained on this lower level class of problems on more complex computers. Hence, the range of computers covered herein is from the PDP-15 through the 1108 and includes the PDP-11, PDP-10, Sigma 2/3, Sigma 5/7, System 86, 360/44, and PDP-K. Secondly, the coding was viewed as an educational exercise to judge the necessity and usefulness of the various instruction sets at the hand coded assembly language level.

As previously noted, the purpose was to compare instruction sets, not computers. Because of this we deliberately excluded some factors that would normally be counted in comparing the performance of several machines. Hence, differences in memory cycle times are not included nor are times required for execution of complex instructions adjusted. These are parameters inherent with the particular technology of implementation of today and are not properly attached to the instruction set.

The measures were number of words required for each program, number of instructions, and the number of memory cycles required to do the program under a specific set of assumptions. This, of course, assumes that main memory will continue to be the main factor in processor cycle speeds. (Perhaps somewhat dubious an assumption in view of what is happening to semiconductor memory prices.)

Comments on Problems 1 through 5

The problems originated with John Cohen and Larry McGowen in PM18-3 (May 3[illegible], 1968) (PDP-Z Distribution). They have since been coded for the PDP-11 by National Information Services, Inc., and are currently being redone by Dave Brown, another consultant contract.

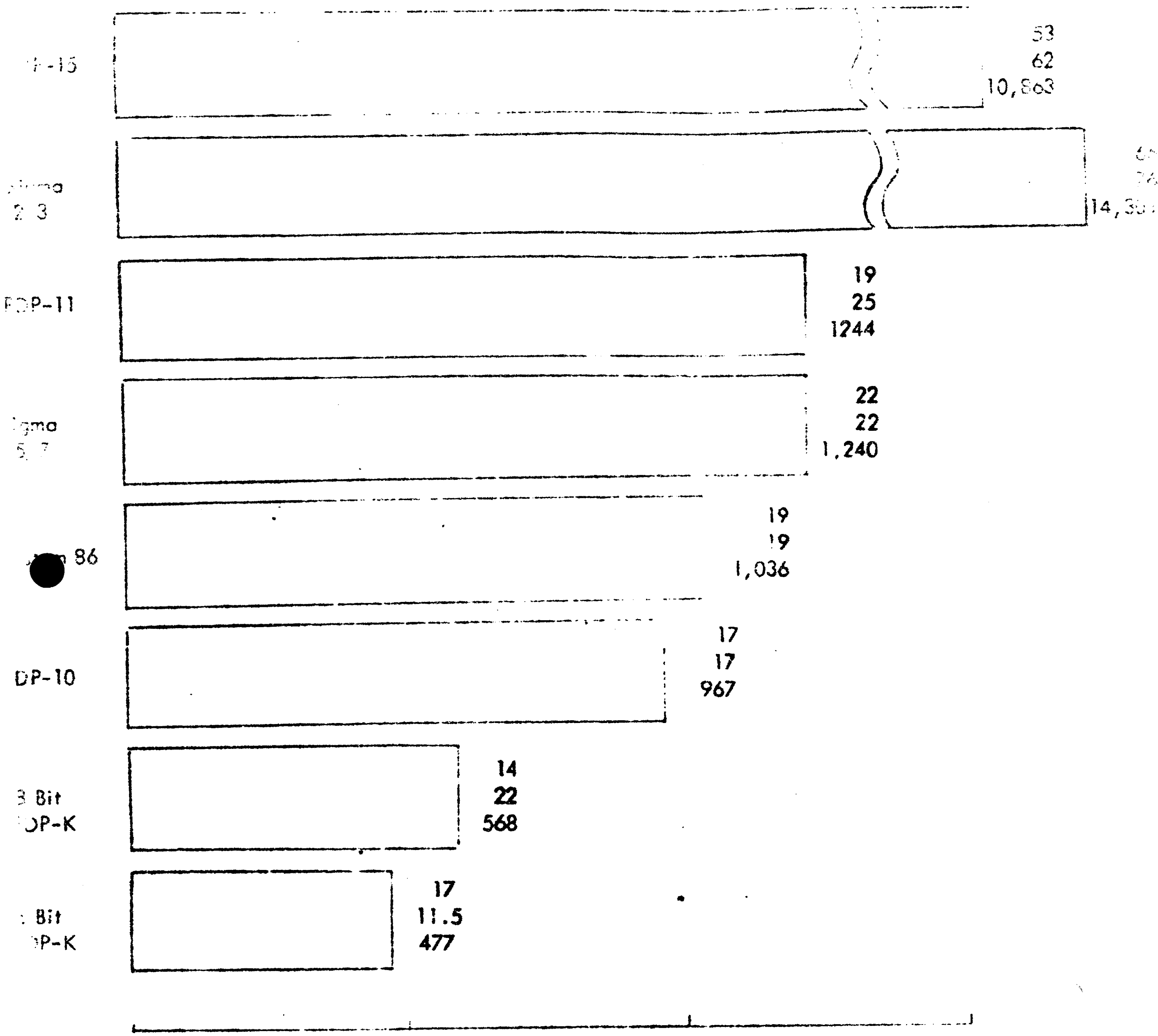
Problem 2 - Multiply is unrealistic as the signs of the number are not checked and dealt with and there is no provision for checking when the multiplier becomes "O".

Problems 3 and 4 are essentially both tolerance checks.

These limitations should be kept in mind when evaluating the results of these problems.

Comments on Problem 1 "Move Characters and Edit"

The graph essentially shows three levels of performance on this problem. The worst performers were the Sigma 2/3 and PDP-15. This was due to their lack of multiple accumulators. In addition, the Sigma 2/3 lacked any byte handling at all, and this cost it many extra memory cycles. The next group had approximately an order of magnitude better performance on this problem - which was obtained from the MOVE BYTE instructions and their multiple AC organization. The performance increase of the PDP-K over this group was solely due to the K's REPEAT instruction.



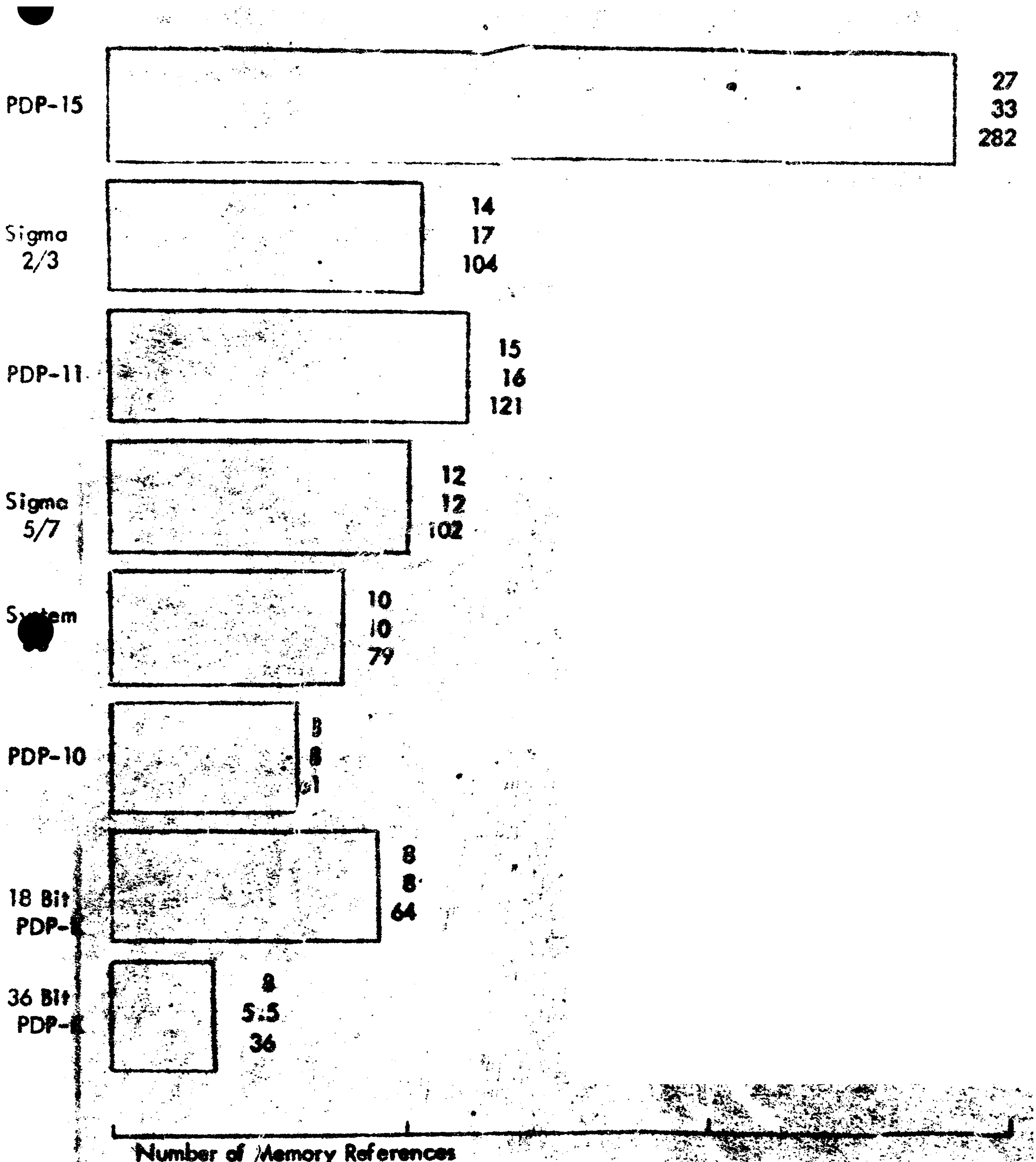
Number of Memory References

PROBLEM #1 Move Characters and Edit

- Top number is number of instructions
- Middle is number of words
- Bottom is the number of memory references

Comments on Problem 2 "Software Multiply"

Performance on this problem was relatively consistant across machines. The PDP-15 was somewhat worse than all the others because it had only 1 AC. The low number of memory references by the PDP-10 was due to its TRN (Test Against Immediate Mask and Skip) instruction.

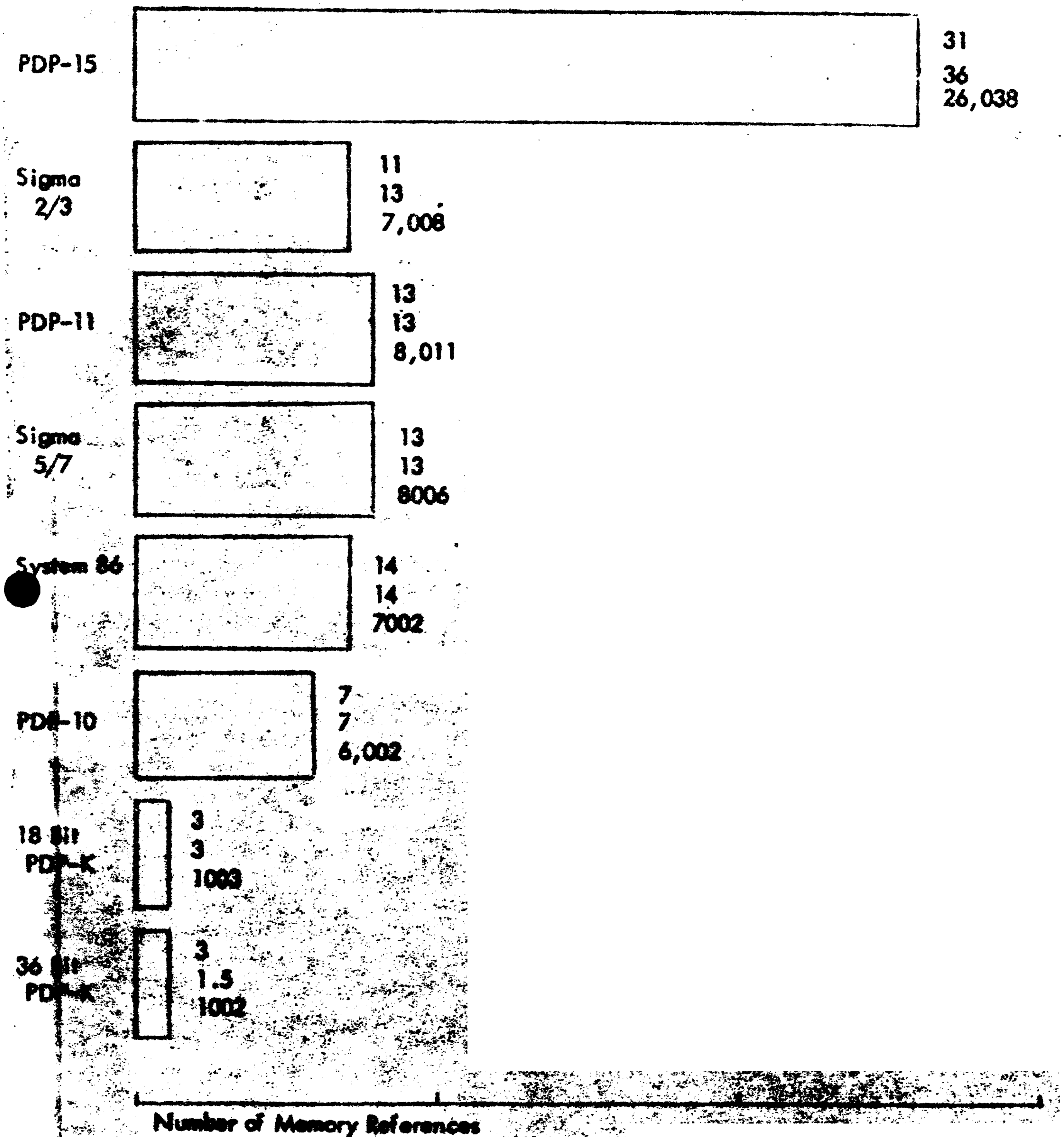


PROBLEM #2 Software Multiply

- Top Number is number of instructions
- Middle is number of words
- Bottom is the number of memory references

Comments on Problem 3 "Tolerance Check"

Though this looks like a "ringer" in the set, it proves the usefulness of K's REPEAT instruction and its COMPARE with LIMITS. These two provide about a 5 fold improvement in performance over the other "high performance" machines. The PDP-10's improvement over the Sigma's and PDP-11 and System 86 is due to its COMPARE AND SKIP instructions.

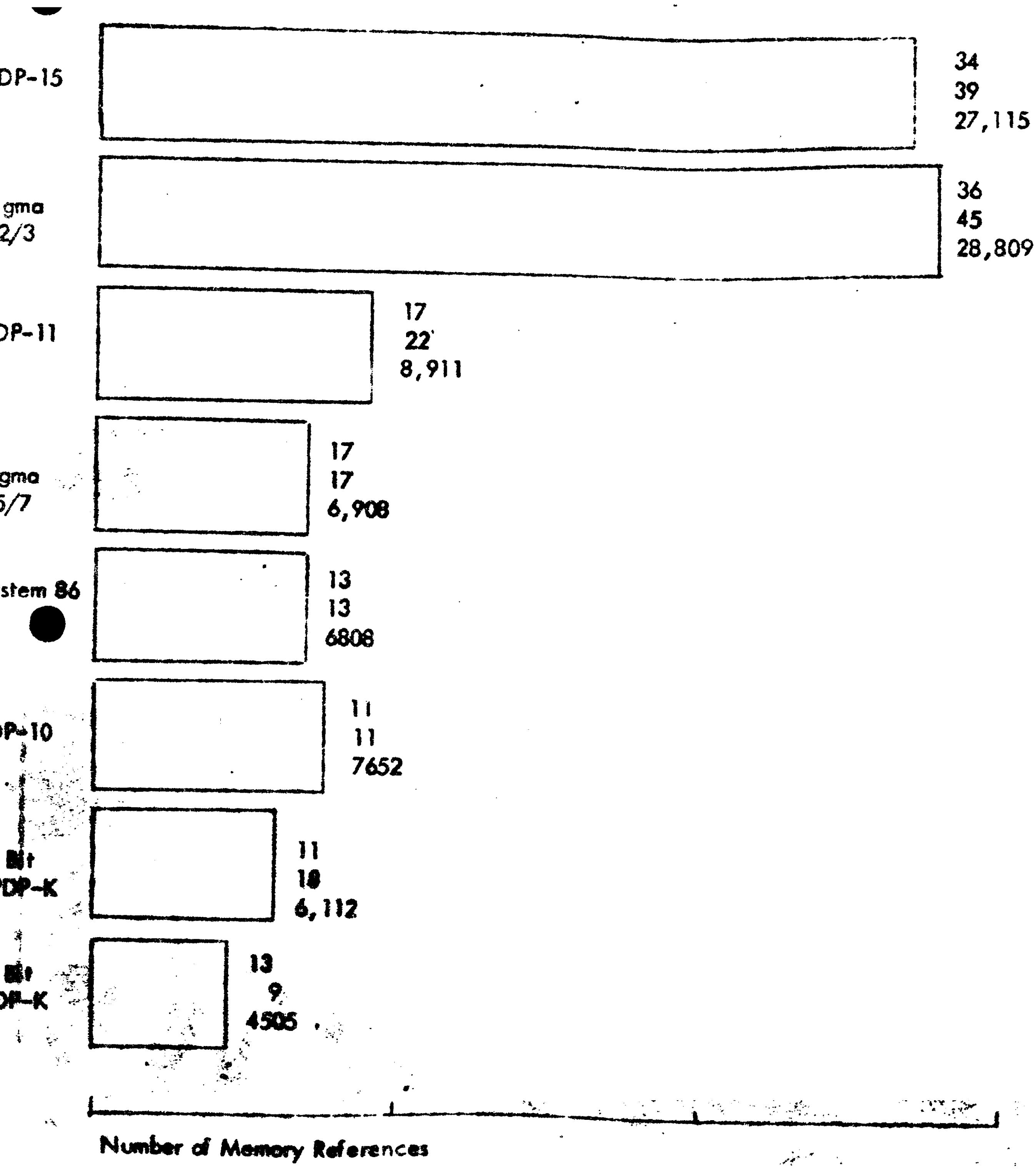


PROBLEM # 3 Tolerance Check

- Top number is number of instructions
- Middle is number of words
- Bottom is the number of memory references

Comments on Problem 4 "Histogram"

PDP-15 and Sigma 2/3 lose again due to lack of byte handling. Again, PDP-K's improvement is due to use of its COMPARE WITH LIMITS instruction.

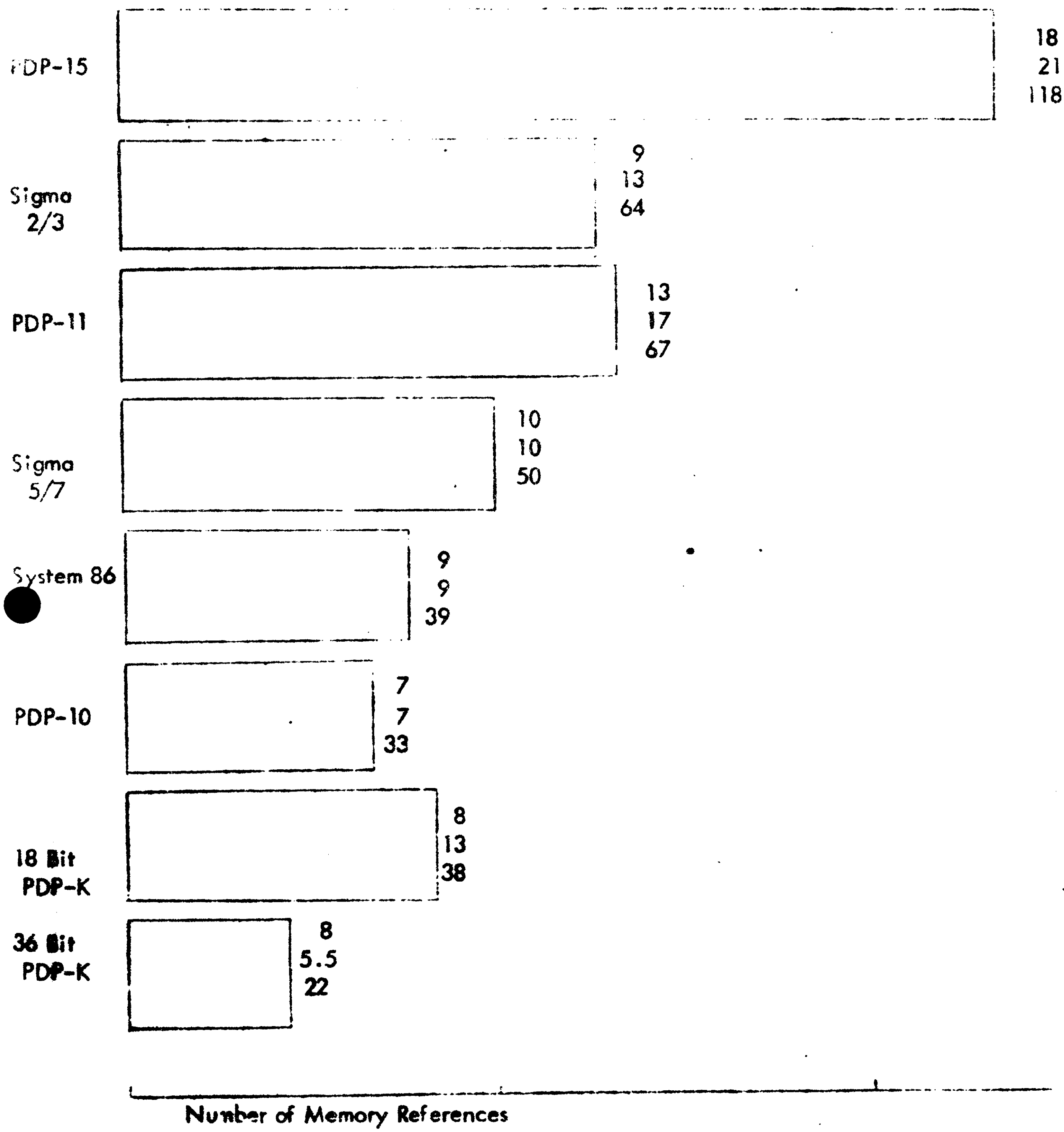


PROBLEM #4 Histogram

- Top number is number of instructions
- Middle is number of words
- Bottom is the number ofme

Comments on Problem 5 "Decimal (ASCII) to Binary Conversion"

Performance is rather consistant on this problem, though PDP-15 loses due to lack of AC's.



PROBLEM #5 Decimal (ASCII) to Binary Conversion

- Top number is number of instructions
- Middle is number of words
- Bottom is the number of memory references