



ETL-0240

**LEVEL** *12*

The design and analysis of a high-production mini-computer system for regriding digital terrain elevation matrices

**AD A100034**

George C. Stockman  
Russell C. Smith  
I-Lok Chang

L.N.K. Corporation  
302 Notley Court  
Silver Spring, MD 20904

**DTIC**  
FLECTER  
JUN 11 1981

OCTOBER 1980

**DTIC FILE COPY**

APPROVED FOR PUBLIC RELEASE DISTRIBUTION UNLIMITED

**81 6 11 059**

U.S. ARMY CORPS OF ENGINEERS  
ENGINEER TOPOGRAPHIC LABORATORIES  
FORT BELVOIR, VIRGINIA 22060



**E**

**T**

**L**



ETL-0240

**LEVEL**

12

The design and analysis of a high-production mini-computer system for regriding digital terrain elevation matrices

**AD A100034**

George C. Stockman  
Russell C. Smith  
I-Lok Chang

L.N.K. Corporation  
302 Notley Court  
Silver Spring, MD 20904

**DTIC**  
REFLECTED  
JUN 11 1981

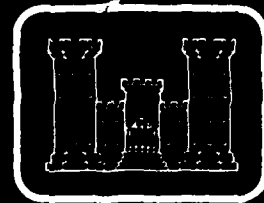
OCTOBER 1980

**DTIC FILE COPY**

APPROVED FOR PUBLIC RELEASE - DISTRIBUTION UNLIMITED

**81 6 11 059**

U.S. ARMY CORPS OF ENGINEERS  
ENGINEER TOPOGRAPHIC LABORATORIES  
FORT BELVOIR, VIRGINIA 22060



**E**

**T**

**L**



Destroy this report when no longer needed.  
Do not return it to the originator.

---

The findings in this report are not to be construed as an official  
Department of the Army position unless so designated by other  
authorized documents.

---

The citation in this report of trade names of commercially available  
products does not constitute official endorsement or approval of the  
use of such products.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 15 ETLA0240	2. GOVT ACCESSION NO. AD-A100 034	3. RECIPIENT'S CATALOG NUMBER (9)
4. TITLE (and Subtitle) THE DESIGN AND ANALYSIS OF A HIGH-PRODUCTION MINI-COMPUTER SYSTEM FOR REGRIDDING DIGITAL TERRAIN ELEVATION MATRICES		5. TYPE OF REPORT & PERIOD COVERED CONTRACT REPORT / FINAL
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) George C. Stockman, Russell C. Smith, I-Lok Chang		8. CONTRACT OR GRANT NUMBER(s) DAAK78-80-C-0022
9. PERFORMING ORGANIZATION NAME AND ADDRESS L.N.K. Corporation 302 Notley Court Silver Spring, Md. 20904		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Engineer Topographic Laboratories Fort Belvoir, Virginia 22060		12. REPORT DATE October 1980
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		13. NUMBER OF PAGES 74
		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report)  Approved for Public Release; Distribution Unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)  Digital Terrain Elevation Matrices Transformation Regridding		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)  This report studies a new algorithm for the regridding and transformation of Digital Terrain Elevation Matrices. The new algorithm uses sequential accesses to mass storage devices rather than the random methods used by the software system currently installed at the Defense Mapping Agency. Due to the simplicity of the new algorithm, minicomputers can be used in place of a large mainframe computer. Four potential minicomputer systems are examined. Benchmark test results are given which show the feasibility of minicomputer use. Recommendations are		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

391718

LB

20. (ABSTRACT continued)

made concerning the minicomputer to select if the new algorithm is to be implemented.

A

PREFACE

This document was generated under Contract DAAK 70-80-C-0022 for the U.S. Army Engineer Topographic Laboratories, Fort Belvoir, Virginia, 22060, by L.N.K. Corporation, 302 Notley Court, Silver Spring, Maryland and submitted as ETL-0240. The Contract Officer's Representative was William Edward Opalski. The authors thank Mr. Opalski and also Mr. Arthur Noma and Dr. Clifford Kottman of the Defense Mapping Agency Hydrographic/Topographic Center for their helpful comments on the draft of this report.

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	<input type="checkbox"/>
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A	

## Executive Summary

The objective of this report is to:

- (1) present an algorithm for regridding and transformation of digital terrain elevation matrices that will efficiently run on a minicomputer,
- (2) benchmark transformation and I/O times on four candidate minicomputer systems,
- (3) make recommendations as to which minicomputer systems will perform most efficiently, and
- (4) recommend modifications to DMAHTC current Mosaicking and Regridding Software System (MARS).

The motivation for the study effort came from the realization that the MARS system, which is implemented on a UNIVAC 1100/81 mainframe, has bottlenecks that result from certain operating system constraints. In looking for a solution to this problem it became apparent that state-of-the-art minicomputer systems, particularly the 32 bit machines, could perform as well or better than the UNIVAC 1100/81, especially if the regridding and transformation functions were performed in a sequential manner rather than the current random manner. A new algorithm was designed with a view toward installing it on a "fast" minicomputer. The new algorithm could be implemented on the UNIVAC 1100/81, however, it would not achieve the same efficiency as it will on a suitable minicomputer.

Section 2 describes the new algorithm and its refinement. The new algorithm rearranges the regridding and transformation task so that sequential accessing methods to mass storage devices can replace the current

random access methods. Due to the resulting simplification of the processing needs the new algorithm can be installed on a minicomputer, allowing easy and inexpensive expansion should there be a future increase in workload.

Section 3 examines the specifications of minicomputer systems which could perform the new algorithm efficiently enough to compete with the UNIVAC 1100/81. Both 16 bit and 32 bit minicomputers are examined. A comparison of relative costs is made. Special hardware options which can significantly improve execution times are noted.

Benchmark tests were run on a subset of the minicomputers presented in Section 3 in order to determine if preliminary estimates of their speeds were correct. In Section 4 the results of these tests are given. I/O transmission speeds are shown to be more than sufficient for the implementation of the new algorithm. The speed at which a minicomputer could perform a compute-bound point-to-point transformation, necessary for the regridding task, shows that the fastest (and most expensive) minicomputers would be necessary if each point in the Digital Terrain Matrix were to be rigorously transformed. However, only a small portion of the points are rigorously transformed by the current software used by DMA. Under these circumstances, all the minicomputers examined can be used for the new algorithm.

Section 5 presents an overview of the study, conclusions, and recommendations. General conclusions regarding the transportation of the software from the UNIVAC 1100/81 to a minicomputer, including estimates of costs, are given. The recommendation is made that a 32 bit minicomputer (PERKIN-ELMER 3240) be chosen for the implementation of the new algorithm. This recommendation is based on the ease of implementation possible with the larger address space of a 32 bit computer when compared to that of a 16 bit computer and the capability of a 32 bit computer in handling future DMA needs.

## Table of Contents

	<u>PAGE</u>
Executive Summary.....	iv
1. Introduction.....	1-1
2. Algorithms.....	2-1
2.1 The Proposed Algorithm.....	2-1
2.2 The Refined Algorithm.....	2-3
2.2.1 Notation.....	2-3
2.2.2 Regridding and Binning.....	2-3
2.2.3 Interpolation.....	2-6
2.2.4 Coordinate Coding.....	2-6
2.2.5 Sampling Grid Generation.....	2-6
2.2.6 Implementation.....	2-8
2.3 Further Analysis of the Refined Algorithm.....	2-8
2.3.1 Assumptions Concerning Hardware.....	2-9
3. Hardware.....	3-1
3.1 The DATA GENERAL Eclipse S/250.....	3-3
3.2 The PERKIN-ELMER 3220 and 3240.....	3-5
3.3 Comparison of Systems.....	3-8
4. Benchmarks.....	4-1
5. Conclusions and Recommendations.....	5-1
5.1 General Conclusions.....	5-1
5.2 Conclusions Regarding Coordinate Transformations....	5-2
5.3 Final Conclusions.....	5-3
 Appendix A -- Study of Transformation Package	
 Appendix B -- Formal Minicomputer Manufacturer's Quotations	

## List of Figures

	<u>PAGE</u>
Figure 1. Flow diagram for the initially proposed algorithm	2-2
Figure 2. Sample Input and Output of Initial MiniMARS Test	2-4
Figure 3. All output grid points will lie in one of $N + 1$ aisles for an $N \times N$ input DTM	2-5
Figure 4. Elevations of transformed output points are interpolated from elevations of input points	2-7
Figure 5. Flow diagram for the refined MiniMARS algorithm	2-14
Figure 6. Peripheral flow diagram for the refined MiniMARS algorithm	2-15

List of Tables

	<u>PAGE</u>
Table 1. Comparative Prices.....	3-7
Table 2. Transformation Benchmarks for MiniMARS.....	4-2
Table 3. Geographic to UTM Times on Four Minicomputers....	4-2
Table 4. Random Reads from Disk.....	4-4
Table 5. Sequential Reads from Disk.....	4-4

## INTRODUCTION

The Defense Mapping Agency Hydrographic/Topographic Center (DMAHTC) has a requirement to convert a large photogrammetric source database to digital terrain elevation matrices. This conversion is currently achieved in large part by scanning raw photogrammetric data with specialized hardware (Universal Automatic Map Compilation Equipment or UNAMACE), the output of which is a matrix of elevation values (Digital Terrain Matrix or DTM). DTM's are combined to form 1<sup>0</sup> squares and stored in a digital database using DMAHTC Standard Format. The process of combining, called mosaicking, involves the regridding and transformation of the DTM's. This is currently accomplished using a complex system of software called the Mosaicking And Regridding System (MARS) developed and implemented at DMAHTC by L.N.K. Corporation in 1978.

MARS, designed to run in batch mode on a UNIVAC 1100/81 computer, has been vital to DMA in handling the conversion of UNAMACE data. However, after working at DMAHTC with the system on a daily basis for two years and gaining an intimate knowledge of the operator interface, machine resource requirements and wall clock time necessary for the regridding and transformation process, L.N.K. determined that some portions of the MARS system were in effect bottlenecks. These bottlenecks are due to constraints imposed by the UNIVAC operating system. In particular, the limiting of the memory address space leads to a fragmentation of the data being processed, thus excessive random accesses to mass storage devices are required.

After collection of statistics on the regridding and transformation process and upon closer examination of both the details of the process and the capabilities of state-of-the-art minicomputers, L.N.K. estimated that the regridding and transformation task could be performed on a dedicated

minicomputer more efficiently than on the UNIVAC 1100/81. In addition, the implementation of a "MiniMARS" would allow for easy expansion to handle any increase in processing needs simply by adding to the number of minicomputers used for the task.

L.N.K. therefore proposed a study effort to:

- (1) Develop a new algorithm for the regridding and transformation process which would significantly decrease the number of random accesses to mass storage devices,
- (2) Examine a selection of minicomputer systems and prove they could efficiently and effectively perform the new algorithm,
- (3) Recommend DMAHTC purchase any one of four select systems, and
- (4) Recommend modifications to the current MARS system which could improve throughput on the UNIVAC 1100/81. These modifications are not an implementation of the new algorithm, but an optimization of the code used in the current MARS.

In the study benchmark tests were devised and implemented on a number of minicomputer systems to determine possible transformation and I/O times for the regridding and transformation task. Using the results from these tests L.N.K. confirmed that state-of-the-art minicomputers (PERKIN-ELMER 3240, DATA GENERAL ECLIPSE S/250, PDP 11/44) can compete with large mainframes such as the UNIVAC 1100/81 when the minicomputers are dedicated to the task at hand. The details of the algorithm, tests, and recommendations are included in this report.

## 2. ALGORITHMS

### 2.1 The Proposed Algorithm

Universally, the regriding and transformation task consists of

- (1) Definition and creation of a target coordinate system grid,
- (2) Input and transformation of input digital elevation data, and
- (3) Creation of the output model.

Regridding and transformation, normally performed using random access methods with the MARS software system, has been restructured in the new algorithm into a series of sequential file processing steps. Each step, due to simplicity, can be efficiently implemented on a minicomputer system.

The proposed new algorithm is a 4 step process. Initially the target coordinate system grid is defined. Each point in this grid ( $x',y'$ ) is transformed to the input model's coordinate system, thus producing a set of quadruples ( $x',y',x,y$ ). These quadruples are then sorted according to the ( $x,y$ ) coordinates. The sorted quadruples are now merged with the input model, interpolating elevation values for ( $x,y$ ) as is currently done with the MARS system. The result of this step is a set of triples ( $x',y',z'$ ) which are in the input model's order. Therefore these triples are resorted according to the ( $x',y'$ ) coordinates, producing a file with the correct order for the output model. Figure 1 illustrates the steps of this process. Note that the transformation from one coordinate system to the other is considered to be done by a "black box", i.e. any transformation function can be used with this system, including those used by MARS.

An elementary version of the algorithm was implemented on a UNIVAC 1108 computer. Initially a small input model (20 x 20 points) was hand-created. A target coordinate system was defined as a small rotation and translation of the input coordinate system. A file of 400 quadruples ( $x',y',x,y$ ) was created.

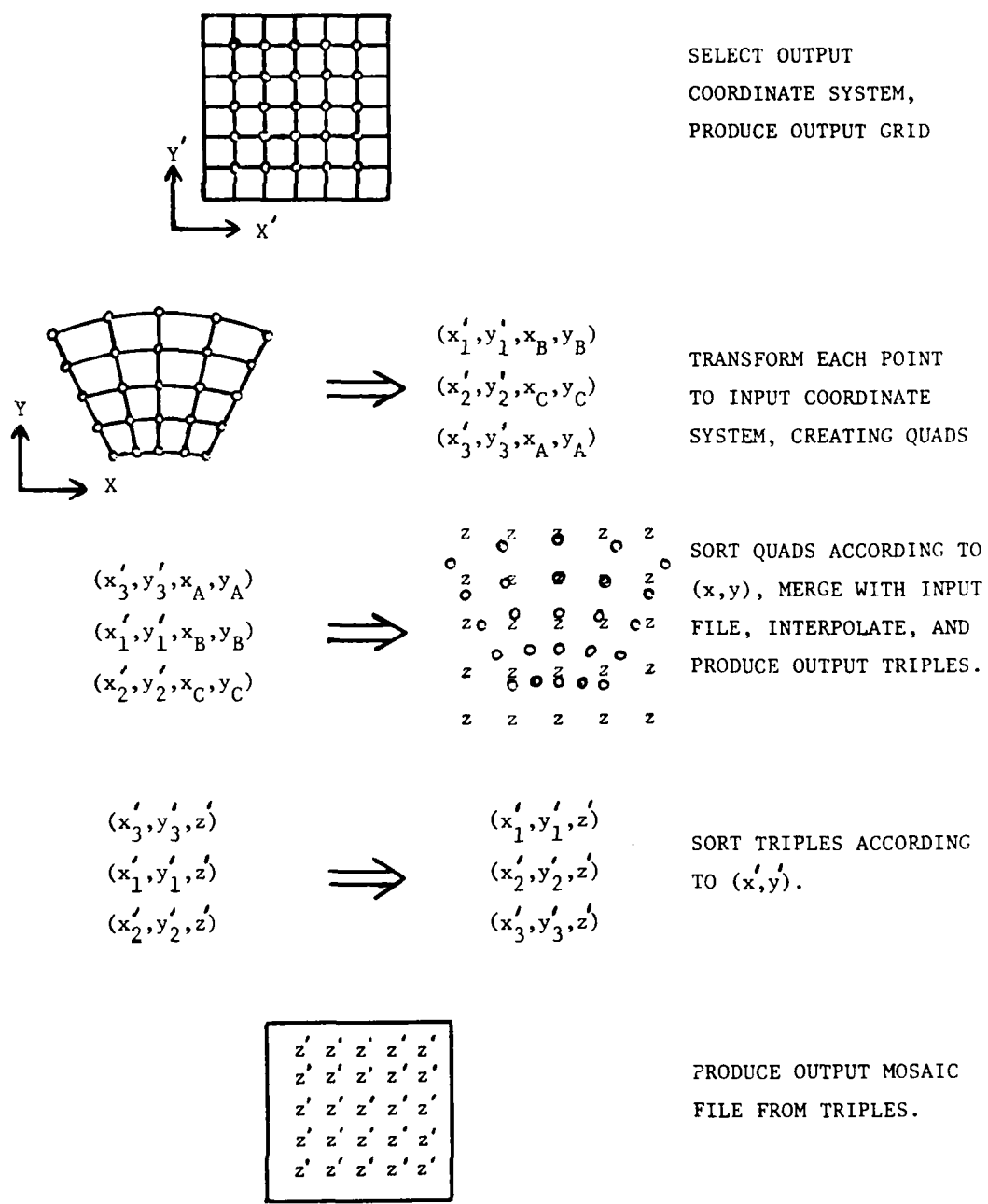


Figure 1. Flow diagram for initially proposed algorithm

The quadruple file was sorted according to the (x,y) coordinates using the UNIVAC system sort routine SORTSDF. The file of quadruples and the input model were then merged, interpolating new elevation values by using a simple four neighbor average. The output of this step (x',y',z') was sorted according to the (x',y') coordinates and then plotted. Figure 2 shows both the input model and the resulting output model. Numerical elevation values are represented by alphanumeric characters. It should be noted that in a real system no elevation data would be thrown away. In this case data was clipped only for uniformity of plot size.

The algorithm thus has been graphically shown to be correct. However on further analysis it was found that by increasing the size of the memory space and using in-core addressing methods the sort steps could be dropped, greatly increasing the throughput of the system. The refined algorithm is detailed below.

## 2.2 The Refined Algorithm

### 2.2.1 NOTATION

As in the proposed system primes are used to indicate points in the output grid coordinates, such as (x',y'). Unprimed quantities are reserved for input grid coordinates, such as (x,y). T(x',y') or T(x',y',z) is the output grid point (x',y') transformed into the input grid coordinate system.

### 2.2.2 REGRIDDING AND BINNING

As Figure 3 shows, output grid points will be transformed into the input Digital Terrain Matrix (DTM) coordinate system so that elevation values can be determined for each one by interpolation. Assuming an N x N input DTM, every transformed grid point must lie in one of N + 1 "aisles" or bins. For a 1024 x 1024 input DTM there are only 1025 bins.

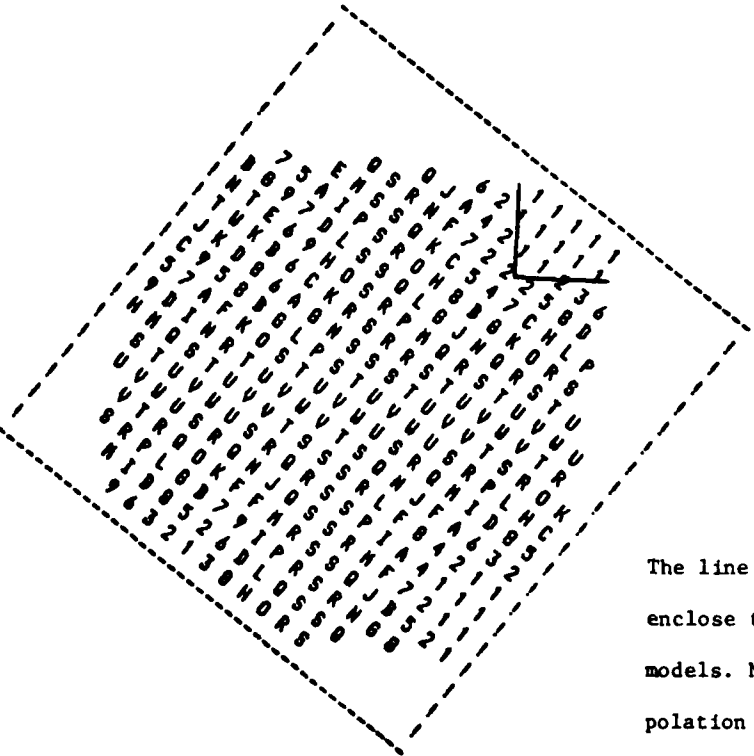
```

-----
! A S S S J S S A K S S Q I 8 2 1 1 1 1 1 !
! A S F F F F S A K S S Q I 8 2 1 1 1 1 1 !
! A S F U U F S A K S S Q I 8 2 1 1 1 1 1 !
! A S F U U F S A K S S Q I 8 2 1 1 1 1 1 !
! A S F F F F S A K S S Q I 8 2 2 2 2 2 2 !
! A S S S S S S A K S S Q I 8 8 8 8 8 8 8 !
! A A A A A A A A K S S Q I I I I I I I I !
! K K K K K K K K S S Q 0 0 0 0 0 0 0 0 0 !
! S S S S S S S S S S S S S S S S S S S !
! U U U U U U U U U U U U U U U U U U U !
! U U U U U U U U U U U U U U U U U U U !
! S S S S S S S S S S S S S S S S S S S !
! Q Q Q Q Q Q Q Q Q Q S S Q 0 0 0 0 0 0 0 !
! I I I I I I I I I I S S Q I I I I I I I I !
! 8 8 8 8 8 8 8 I Q S S Q I 8 8 8 8 8 8 8 !
! 2 2 2 2 2 2 8 I Q S S Q I 8 2 2 2 2 2 2 !
! 1 1 1 1 1 2 8 I Q S S Q I 8 2 1 1 1 1 1 !
! 1 1 1 1 1 2 8 I Q S S Q I 8 2 1 1 1 1 1 !
! 1 1 1 1 1 2 8 I Q S S Q I 8 2 1 1 1 1 1 !
! 1 1 1 1 1 2 8 I Q S S Q I 8 2 1 1 1 1 1 !
-----

```

ORIGINAL INPUT  
MODEL

RS&T  
TRANSFORMATION  
R = 30°  
S = 1  
T = (=8,-4)



MODEL AFTER TRANSFORMATION  
TO OUTPUT COORDINATE SYSTEM

The line segments at right angles enclose the same region in the two models. Note that due to interpolation the elevation values are slightly changed at region borders.

Figure 2. Sample Input and Output of Initial MiniMARS Test

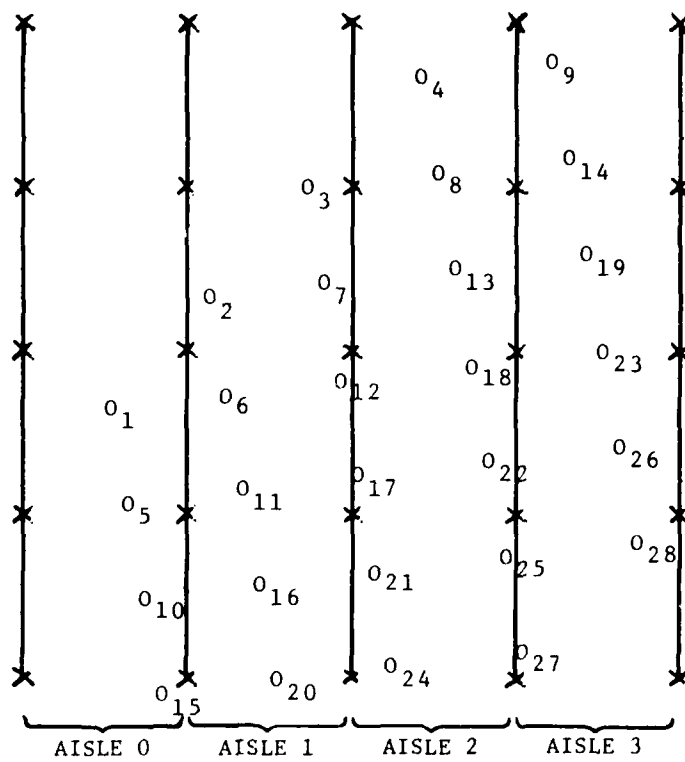


Figure 3. All output grid points will lie in one of  $N+1$  "aisles" for an  $N \times N$  input DTM.

### 2.2.3 INTERPOLATION

During interpolation, the elevation of each output grid point ( $0_1, 0_2, \dots, 0_{28}$  in Figure 3) will be determined from the elevations of only 4 points in the input DTM, which lie on only two profiles. Figure 4 shows the 4 "neighbors" used to compute the elevation at point  $T(x', y')$ . If the input grid points are never more than 128 meters apart at true scale a fraction of 6-bit precision for  $T(x', y')$  will yield 2 meter positional accuracy in the  $(x, y)$  space.

### 2.2.4 COORDINATE CODING

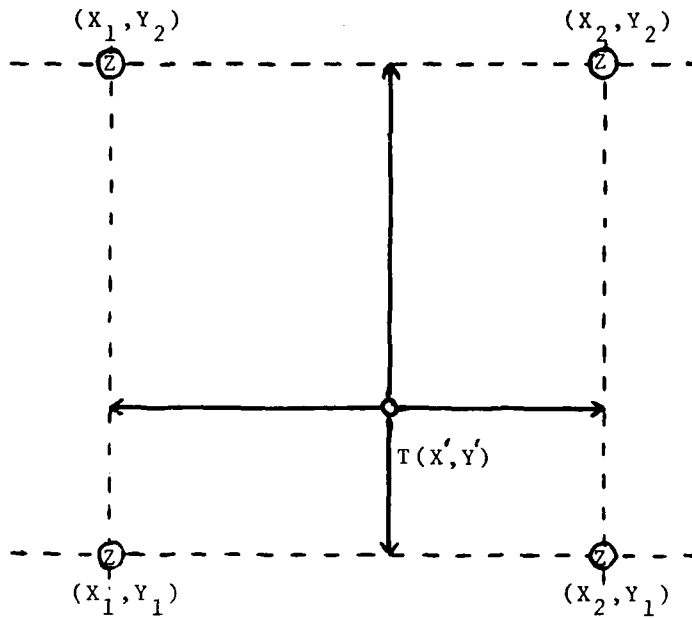
Using 16 bits overall and 6 bits for a fraction,  $T(x', y') = (x, y)$  will be coded as follows

$$x \approx D_{x1}D_{x2} \dots D_{x10} \cdot F_{x1}F_{x2} \dots F_{x6}$$
$$y \approx D_{y1}D_{y2} \dots D_{y10} \cdot F_{y1}F_{y2} \dots F_{y6}$$

The consequences of 2 meter accuracy at up to 128 meter spacing is a limit on the model size to 1024 x 1024 if the above 16-bit coordinates are to be used. However, the above bit pattern can be varied as is needed. Using fewer fraction bits allows a larger DTM to be manipulated.

### 2.2.5 SAMPLING GRID GENERATION

The output grid points will be generated and transformed into the input coordinate space. During generation each point will be represented by a quadruple  $(x', y', x, y)$ , where  $(x, y) = T(x', y')$ , which will immediately be placed in the correct aisle for later sequential merging with the input



ELEVATION AT  $T(X', Y')$  IS INTERPOLATED FROM THE ELEVATIONS AT  $(X_1, Y_1)$ ,  $(X_2, Y_1)$ ,  $(X_1, Y_2)$ , AND  $(X_2, Y_2)$ . IF INPUT SPACING IS LESS THAN OR EQUAL TO 128 METERS, A 6 BIT FRACTION WILL YIELD 2 METER ACCURACY.

Figure 4. Elevations of transformed output points are interpolated from elevations of input points.

DTM. The rough algorithm is as follows:

- (A1) Get input DTM specs from the header and define the output grid and transformation T.
- (A2) Generate and bin each point of the output grid
  - (A2.1) For each point  $(x',y')$  produce the quadruple  $(x',y',x,y)$  where  $(x,y) = T(x',y')$  and where  $x \approx D_{x1}D_{x2}\dots D_{x10} \cdot F_{x1}F_{x2}\dots F_{x6}$  and  $y \approx D_{y1}D_{y2}\dots D_{y10} \cdot F_{y1}F_{y2}\dots F_{y6}$
  - (A2.2) Place each quadruple  $(x',y',x,y)$  into aisle A which is  $D_{x1}D_{x2}\dots D_{x10}$  in binary.

#### 2.2.6 IMPLEMENTATION

No aisle can receive more than  $1024 \sqrt{2}$  quadruples (the square root value comes from the diagonal of the matrix) hence 8 times this (about 15000) bytes of storage are needed for each aisle. In main memory we can statically allocate 1025 blocks of size 25 quadruples (i.e. 200 bytes) each. When full, each 25 quadruple block will be output to a known track storing the appropriate aisle. There will thus be about  $2^{20}/25$  writes (about 40000). This will require  $40000 \times 43$  ms average access time plus  $2^{23}$  bytes /  $(8 \times 10^5$  bytes/second transfer rate) or about 1730 seconds, approximately 1/2 hour. A refined algorithm will now be given which will reduce the number of disk writes, and hence the execution time, by half.

#### 2.3 Further Analysis of the Refined Algorithm

An in depth analysis of the refined algorithm using data on off-the-shelf minicomputer systems is given on the following pages. Though the DATA GENERAL S/250 was used for the basis of the times given, all the machines examined in Section 3 have similar I/O speeds. Therefore the use of the DATA GENERAL times should not be construed as a recommendation of the S/250.

### 2.3.1 ASSUMPTIONS CONCERNING HARDWARE

The refined algorithm which follows refers to general specifications for high speed minicomputer systems, for example the DATA GENERAL S/250 or the PERKIN-ELMER 3240. Memory size necessary for efficient implementation of the algorithm should be at least  $2^{18}$  bytes (256K bytes). Peripheral equipment also needed include two independent disk drives (i.e., capable of performing operations simultaneously) and two 1600BPI 9-track tape drives. The disk drives assumed each have five surfaces and 815 cylinders. Each track (intersection of one surface with a cylinder) contains 24 sectors of 512 bytes. The average access time of the disk drives is an important parameter and should be close to the 43 millisecond access time of the DATA GENERAL drives. The tape drives, though not as important with respect to speed, are assumed to be capable of reading and writing at 75 inches/second.

Step 1. Generate output grid, transform, and "aisle sort" to disk.

Algorithm

Read header of input DTM and define output grid.

Generate and transform output grid points (x',y') to get quads (x',y',x,y) with (x,y) = T(x',y').

Output quads in blocks of 512 bytes, or 64 quads, to one of 256 cylinders on disk according to first 8 bits of x. (i.e. each cylinder would store 4 aisles of data)

Main Memory Data Structures

256 block buffers @ 512 bytes each (128K bytes)

3 output buffers @ 512 bytes each (1.5K bytes)

256 block headers @ 4 bytes each (1K bytes)

Disk Structure

bin K stored on cylinder K,  $0 \leq K \leq 255$

K selected as first 8 bits of x in quad

60,000 bytes per cylinder allows 7,000 quads max per 4 aisles

Timing for  $2^{20}$  points assuming no I/O overlap

Transform: 3 to 45 minutes

Output: 8 Mbytes/512 = 16K blocks

16K blocks x  $50 \times 10^{-3}$  seconds each  $\approx$  800 seconds = 13 minutes

Step 2. Collate quads with input DTM profiles and sort on x' on output

Algorithm

Read first 5 input DTM profiles.

Read next block of quads in one of 4 aisles between the 5 profiles.

For each quad  $(x',y',x,y)$  use  $(x,y)$  to interpolate  $z'$ . Then place  $(x',y',z')$  in output block buffer according to the first byte of  $x'$ . When the current 4 aisles are exhausted move on to the next 4 input DTM profiles.

#### Main Memory Data Structures

Tape: 10 input profile buffers @ 2,050 bytes each (20K bytes)  
256 block buffers @ 512 bytes each (128K bytes)  
256 block headers @ 4 bytes each (1K bytes)

Disk out: 3 output buffers @ 512 bytes each (1.5K bytes)

Disk in: 3 input buffers @ 2,048 bytes each (6K bytes)

#### Disk Structure

Input: quad file as in Step 1 output

Output triples

bin  $K$  stored on cylinder  $K$ ,  $0 \leq K \leq 255$

$K$  selected as first byte of  $x'$

85 triples per 512 byte block, hence 10,200 triples per cylinder

this allows 10 profiles @ 1,020 points each; typically only

4 will exist since  $x'$  has only 10 bits of integer

#### Timing for $2^{20}$ points assuming no I/O overlap

input from tape 1000 blocks @ 2" each

2000" @ 75ips = 30 seconds

input from disk<sub>1</sub>

4K blocks (of 2048 bytes) x  $50 \times 10^{-3}$  seconds each = 200 seconds

output to disk<sub>2</sub>

12K blocks (of 512 bytes) x  $50 \times 10^{-3}$  seconds = 600 seconds

main memory processing (estimated interpolation time)

≈ 180 seconds

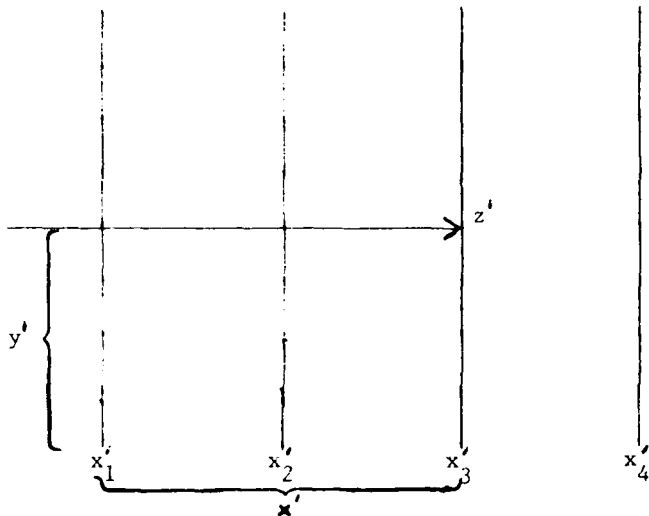
TOTAL TIME ≈ 1,000 seconds

Step 3. Final Model Output

Algorithm

Read entire next bin (cylinder) of triples and assemble up to 10 profiles of  $z'$  values by selecting the profile with  $x'$  and the point position with  $y'$ .

Repeat for all bins.



Place  $z'$  in position addressed by  $x'$  and  $y'$

Main Memory Data Structures

output buffers for output tape

20 profiles x 1,500 points x 2 bytes each = 60,000 bytes

for 2 sets of buffers

input buffers for triples

2 buffers @ 2,048 bytes each

Timing assuming no overlap for  $2^{20}$  points

writing output tape  $\approx$  30 seconds

in core  $\approx$  200 seconds (very conservative estimated time)

input of 3K blocks of 2,048 bytes each

$3K \times 50 \times 10^{-3}$  seconds = 150 seconds

TOTAL TIME  $\approx$  380 seconds

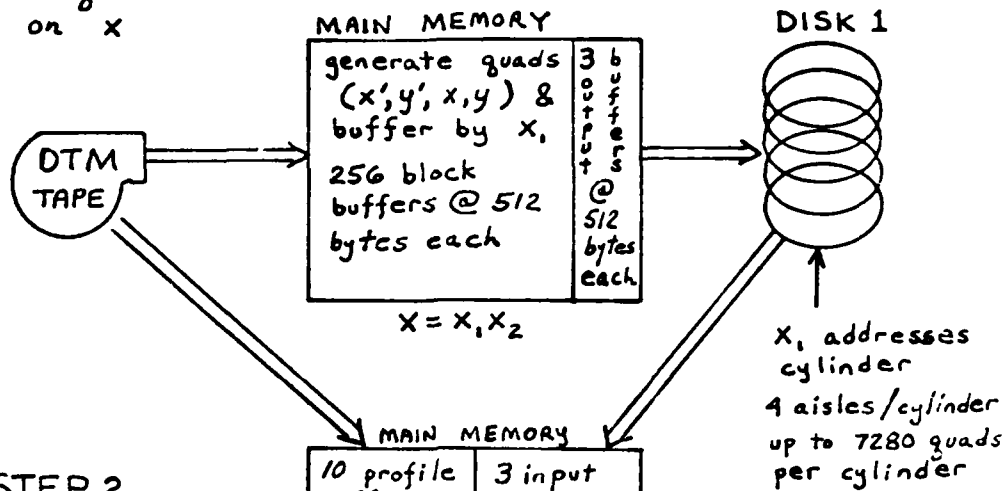
TOTAL TIMES FOR THE 3 STEPS

Step 1.	17 to 58 minutes*	(clear bottleneck)
Step 2.	16 minutes	
Step 3.	<u>7 minutes</u>	
TOTAL	40 to 80 minutes	

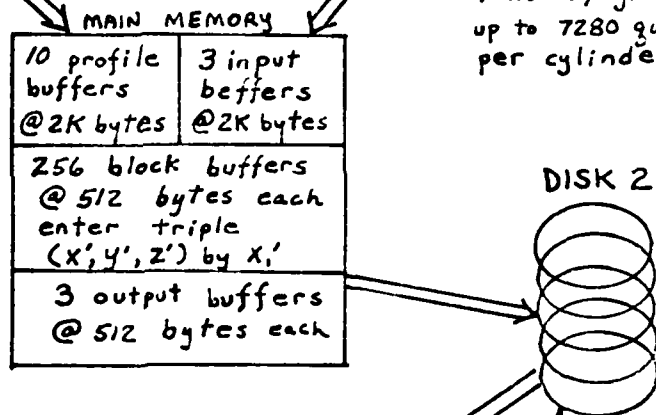
\*high value for rigorous point-to-point transformation; low value for maximum interpolation

Figures 5 and 6 illustrate the refined algorithm. The sort steps have essentially been eliminated through the use of direct addressing methods (both main memory and disk). The refined algorithm requires a larger ( $256K = 2^{18}$ ) memory space than is normally directly addressable by a 16 bit mini-computer.

STEP 1  
Generate and presort quad file on  $x$



STEP 2  
Collate, interpolate, presort on  $x'$



STEP 3  
Full sort of output grid & formation of output DTM

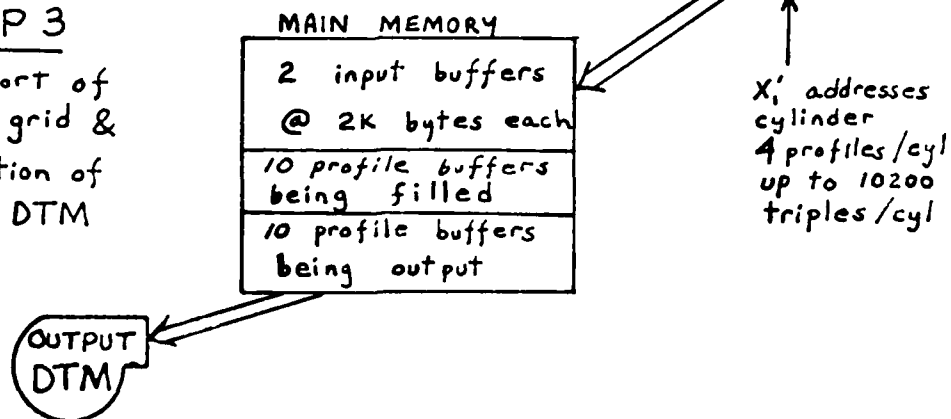


Figure 5. Flow diagram for the refined MinimARS algorithm

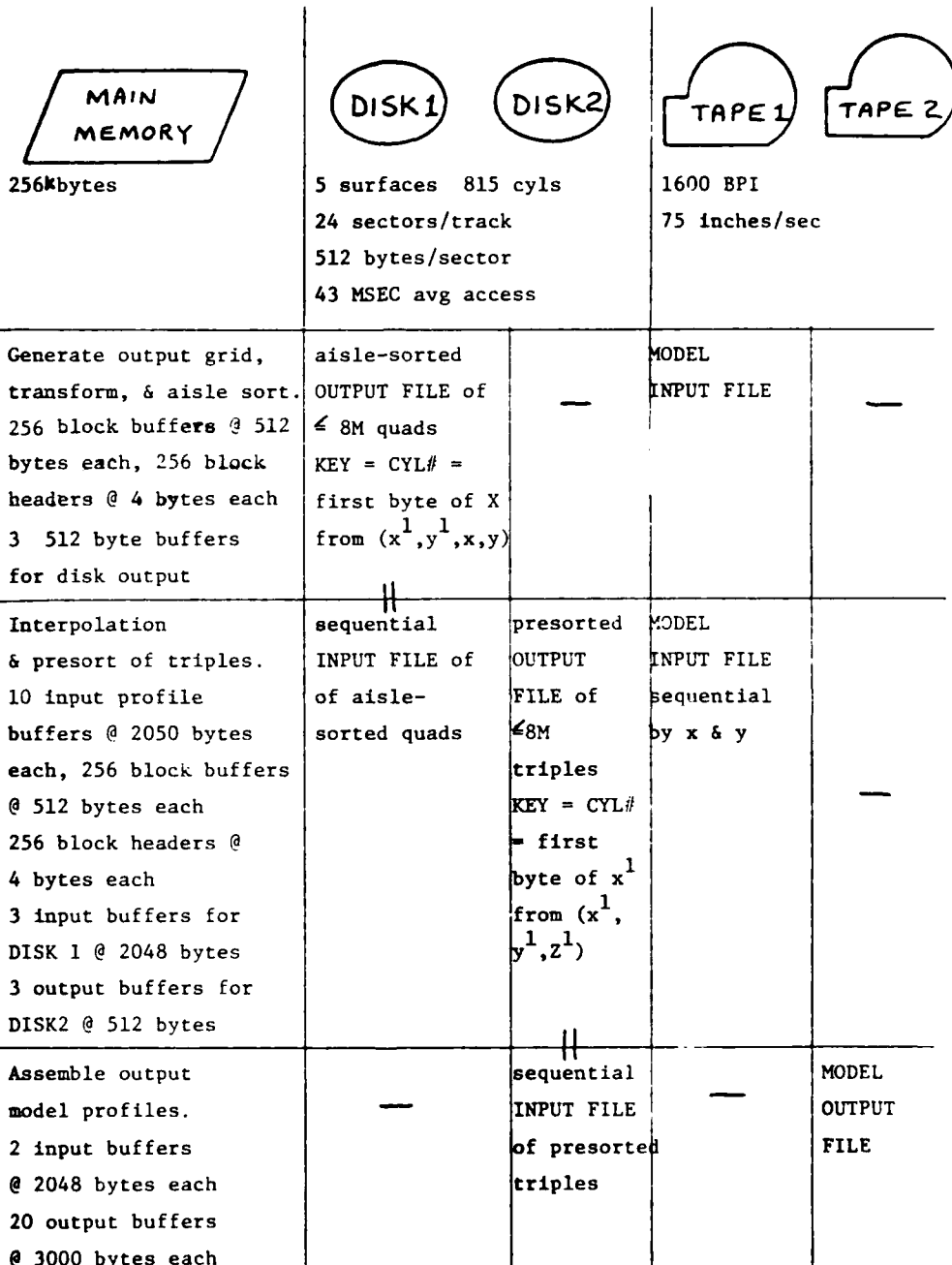


Figure f. Peripheral flow diagram for the refined MiniMARS algorithm

3. HARDWARE

The refined regridding and mosaicking process requires a fair amount of main memory (256K bytes) in order for it to perform fast enough to be more efficient than the methods used in the MARS software system on the UNIVAC 1108. The basic requirements of the minicomputer hardware system are:

- 1) 256K byte main memory (minimum)
- 2) Floating point instructions (hardware, not software interpreted)
- 3) 2 fast disk drives (average access time approximately 45 milliseconds), capacity of each drive at least 50M bytes
- 4) Two 1600 bpi 9 track tape drives, 75 inch per second read/write speed (or faster).

In addition, some method of producing hardcopy output for monitoring the regridding and mosaicking process, such as a printing console terminal, and software comprising a FORTRAN compiler and text editor are essential.

Table 1 gives a price breakdown of 4 off-the-shelf minicomputer systems which meet these requirements. The prices stated are based on the formal manufacturer's quotes contained in Appendix B. The formal quotes, compiled by sales representatives of the companies DATA GENERAL, PERKIN-ELMER, and DIGITAL EQUIPMENT contain certain discrepancies in system configuration which are outlined on the quotation sheets. The discrepancies can increase the bottom line costs for the systems by as much as 20%. Table 1 should be used as the cost guideline in place of the Quotation sheets. All prices in Table 1 reflect the manufacturer's GSA discount, 15% for DATA GENERAL and DIGITAL EQUIPMENT, 21% for PERKIN-ELMER.

The systems examined were felt to be adequate for the task at hand since each could perform the approximate transformation on a set of points

with sufficient speed to compete with the performance of the UNIVAC 1108. DIGITAL EQUIPMENT Corporation's PDP11 systems were not studied as thoroughly as the other three systems, however, because it was found that they could not meet the timing specifications if rigorous transformation was to be performed on more than a small percentage ( $\approx 1\%$ ) of a given set of points. Based on manufacturer's literature the arithmetic logic is much slower than that of the other systems studied, but they can still perform to specs provided that few rigorous transformations are required (It should be noted that the current MARS system rigorously transforms only about 0.5% of the set of points, the remainder being approximately transformed). The sections following provide a closer look at the DATA GENERAL Eclipse S/250 and the PERKIN-ELMER 3220 and 3240 minicomputers. Conclusions and recommendations are given in Section 5.

### 3.1 The DATA GENERAL Eclipse S/250

The DATA GENERAL Eclipse S/250 is a general purpose 16 bit minicomputer with a large and varied instruction set. Its optional floating point processor has the capability of 64 bit precision, necessary for the transformations used and can preserve least significant digit accuracy over many computations. The S/250 also has a Writable Control Store option which could prove to be highly useful for improving the speed of certain time intensive computations. With this option installed, frequently used functions or subroutines (such as the trigonometric functions SIN, COS, and TAN) could be coded to work much faster than that possible with assembly language. The resulting increase in execution speed could be significant (see Section 4). In general, the ability to microcode a computer allows one to tailor the hardware to specific needs.

DATA GENERAL manufactures its own disk drives. The drives considered have a capacity of about 50M bytes broken down into 5 surfaces each with 815 tracks. Each track contains 24 sectors of 512 bytes. Though the desired disk addressing would be in powers of two, it is not felt that the effect of non-power of two addressing present on these drives will be significant. These disk drives are high speed with an average access time of 43 milliseconds.

The DATA GENERAL tape drives are industry compatible 9 track 800/1600 bpi 75 inch/second, capable of working with the high speed Burst Multiplexor Channel option of the S/250. The Burst Multiplexor Channel permits very fast direct memory access for both the tape and disk drives. Due to its relative low cost ( $\approx 2\%$  of total system cost) it is felt that this option should be included if the S/250 is selected for the MiniMARS project.

One interesting aspect of the S/250 is the optional built-in array processor. The regriding and transformation task lends itself to implementation on an array processor since the transformation is to be performed

on a set of points, not unlike operating with a vector of elevation values. This processor, installed in the main CPU cabinet, will recognize special instructions which allow the same operation to be performed on a set of values. The possible increase in execution speed is similar to that obtained with the Writable Control Store. The array processor permits a floating point precision of 32 bits, adequate for many parts of the regridding and mosaicking task.

### 3.2 The PERKIN-ELMER 3220 and 3240

The PERKIN-ELMER 3220 minicomputer and its upgrade, the 3240, are science oriented 32 bit computers. Both machines offer a floating point processor option with 64 bit precision. The major difference between the PERKIN-ELMER floating point processor and those of other manufacturers (including DATA GENERAL) is the rounding method used after performing a floating point operation. Called R\* (R-STAR) rounding, the method can guarantee results at least as accurate as those obtained by truncation or one way rounding over long series of computations. In actual fact, R\* results are usually more accurate and could prove useful for the transformation software.

Both the 3220 and 3240 have multiway memory interleaving which can increase the apparent memory speed with a resultant decrease in execution time. The PERKIN-ELMER 3240 has a Writable Control Store option which can cause significant improvement in execution speed, to the extent that transformation times on the 3240 with microcoded transcendental functions are comparable to those obtained from a UNIVAC 1108 (see Section 4).

The disk drives used by PERKIN-ELMER are supplied by Control Data Corp. They are similar to the DATA GENERAL drives in layout but have a slightly higher capacity (67M bytes) and slightly faster access time (38ms). The higher speed may contribute about a 6% decrease in the I/O times when compared to those of DATA GENERAL. The industry compatible tape drives which come with the PERKIN-ELMER machines (as configured) are virtually identical to those of DATA GENERAL.

PERKIN-ELMER minicomputers do not have a built-in array processor option. Instead, the array processors offered by PERKIN-ELMER are in free standing cabinets and are produced by Floating Point Systems. These processors come in a variety of configurations including models with either 32 or 64 bit precision.

Certain parts of the transformation algorithms require a high degree of precision. Should the array processor option be desired, selecting one with 64 bit precision would allow more of the transformation algorithm to be used in a vector of values at one time than if a 32 bit precision model is chosen. The difference in cost between the 32 bit and 64 bit Floating Point Systems array processors would have to be taken into consideration before any firm decision could be made (FPS recently announced their 64 bit array processor. The 32 bit processor is in the \$100K range. The 64 bit processor is nearly double that.).

Table 1.

## COMPARATIVE PRICES

ITEM	DATA GENERAL S/250	PERKIN-ELMER 3220	PERKIN-ELMER 3240	DEC 11/44
CPU + 256K Byte Memory	\$34,425	\$28,440	\$87,690	\$23,545
Floating Point roc.	\$5,266	\$4,424	\$7,505	\$2635
High Speed (1) Disk Drive (2)	\$18,530 \$18,530	\$12,008 \$12,008	\$12,008 \$12,008	\$21,845 \$16,405
Tape Drive (1) High Speed (2)	\$13,175 \$9,605	\$15,089 \$7,603	\$15,089 \$7,603	\$17,170 \$10,880
Printer	\$2,252	\$3,476	\$3,476	\$1445
Other: Cache, Power Supply, Cabinets, Clock, I/O Interface, etc.	(cabinet) \$2,600 (BMC) \$2,678 (cache) \$3,570	(cabinet) \$869 (channels) \$2,765 (cache) \$2,765 (power) \$1,343 (clock) \$711	(channels) \$2,765 (cache) \$5,925	(cabinet) \$1105 (expansion) \$2720 (interface) \$731 (cable) \$51 (clock) \$697
Software	\$4,400	\$12,600	\$12,600	\$8800
TOTAL	\$115,031	\$104,101	\$166,669	\$107,978

\*NOTE\* The PERKIN-ELMER software costs reflect their pricing for their very powerful optimizing FORTRAN compiler. Duplication costs (for multiple systems) are one tenth of initial costs (for example, \$12,600 for the first minicomputer system, \$1260 for each next system.)

### 3.3 Comparison of Systems

The costs of the systems previously described are roughly equivalent. The 16 bit DATA GENERAL Eclipse S/250, DEC's PDP11/44, and the PERKIN-ELMER 3220 and 3240 are within the \$100K to \$200K range mentioned in the initial proposal. The PERKIN-ELMER 3240, while considerably more expensive than the other systems, has high speed capabilities approaching those obtainable with a third generation machine like the UNIVAC 1100/81. All four systems can perform the proposed regridding and mosaicking task within the time constraint if the approximate transformation is used for the major portion of the points as is currently done with the MARS system. The decision on which machine to choose should be based not only on the capability to perform the regridding and mosaicking task, but also on the capability to perform future tasks, the ease with which the system can be integrated into the existing DMA framework, and the reputation of the manufacturer(s) for delivery and support.

The DATA GENERAL S/250 is the same basic machine used in the UNAMACE equipment. As such, DMA has personnel who are familiar with the inevitable idiosyncrasies of a particular hardware system. However, the S/250 is only a 16 bit machine. Future work which DMA may desire to perform on the chosen system could possibly not be performed adequately on a 16 bit machine. On the other hand, since map oriented operations can quite often be realized as operations on matrices of elevation (or other) values, the built-in array processor of the S/250 could prove to be very useful for quick operations on a vector of values. The S/250 is a fast and powerful 16 bit minicomputer backed by a large experienced corporation.

The PERKIN-ELMER 3220 (or 3240) was designed as a scientific use 32 bit machine. Its instruction set and hardware configuration are oriented towards

manipulation of numerical data in an efficient manner. In addition the 3220 has a 32 bit architecture. Future DMA needs could be implemented on this machine with minimum modification. PERKIN-ELMER's optimizing FORTRAN compiler produces object code which approaches that of an experienced programmer. The "deluxe" 3240 minicomputer is the most powerful of the machines reviewed for this report. Its extremely high speed operations combined with a large memory address space would permit easy future software expansion according to DMA needs. FERKIN-ELMER is a large well known firm involved in many different fields.

DIGITAL EQUIPMENT's PDP11/44 (or similar models) are 16 bit minicomputers with general purpose instruction sets. The speed of this machine is adequate when an approximate transformation is used. Being one of the first minicomputers introduced, the PDP line has a proven architecture for a variety of tasks. In addition, the PDP is familiar to more programmers than the other machines, largely due to its presence at many universities. DEC is the largest minicomputer manufacturer and is known for its support services.

As this report was in the final stages of being written DEC announced a less expensive version of their VAX line of computers. The VAX is an extremely powerful 32 bit machine with a large memory address space. Before any decision is made on the type of machine to be used for the MiniMARS system we feel that this machine should be examined.

#### 4. BENCHMARKS

The MiniMARS algorithm relies on the ability of a minicomputer system to perform certain operations such that when comparisons are made the time required on a UNIVAC 1100/81 and the minicomputer are not radically different. With this requirement in mind, benchmark tests were performed on a UNIVAC and on four different minicomputers. Table 2 shows execution times for three different transformations performed on  $10^4$  points and the extrapolated time estimates for  $10^6$  points. The tests were performed on a UNIVAC 1108 and on a UNIVAC 1180. The minimal driver and transformation routines were then transported to a PDP11/45, P/E 3220 and 3240, and DG S/250 machines. Table 3 gives the times for  $10^4$  points and extrapolated values for  $10^6$  points. As can be seen, the times for the obsolete PDP11/45 are inadequate for rigorous transformation of all points of a model.

The PERKIN-ELMER machines show two times. These result from using a development compiler and an optimizing compiler. The times for the 3220 are within a factor of three of those of the UNIVAC 1100/81. The times for the 3240 are within a factor of two of those obtained from the UNIVAC systems, demonstrating the speed with which state-of-the-art minicomputers can perform. The lower estimate of the 3240 machine was obtained by using a compiler which places certain operations in writeable control store (in this case, sines and cosines). The use of microcoded instructions can increase performance by as much as 30%.

In order to test I/O speeds for disk accessing, a test was run on a PDP11/45 running Bell Laboratory's UNIX operating system. The disk used was an RK05 equivalent with 70 ms average access time. This increase in access time over the proposed access time should be taken into consideration. Table 4 gives the results of 4 tests of pseudo-random reads of

Table 2.  
Transformation Benchmarks For MiniMARS

Transformation	UNIVAC 1108		UNIVAC 1100/81 *	
	CPU Time for 10 <sup>4</sup> Points	Extrapolated 10 <sup>6</sup> Points	CPU Time for 10 <sup>4</sup> Points	Extrapolated 10 <sup>6</sup> Points
Geographic to UTM	7.1 seconds	12 minutes	8.5 seconds	14 minutes
UNAMACE to Geographic	13.8 seconds	23 minutes	16.5 seconds	28 minutes
UNAMACE to UTM	20.1 seconds	35 minutes	24.6 seconds	41 minutes

\* -- These time were obtained while running the tests in "DEMAND" mode on the machine at DMAHTC. The algorithm used for computing execution time has been improved on the 1100/81. This probably accounts for the longer execution times than on the UNIVAC 1108.

Table 3.  
Geographic to UTM Times on Four Minicomputers

	DATA	PERKIN *		PERKIN *		DEC
	GENERAL S/250	ELMER 3220		ELMER 3240		PDP 11/45
CPU Time for 10 <sup>4</sup> Points (seconds)	20	30	23	17.6	12.8	132
Extrapolated for 10 <sup>6</sup> Points (minutes)	33.3	50	36.3	28	21.3	220

\*-- Two times are shown for the Perkin-Elmer machines. The longer time in each case was obtained using a "development" compiler, the shorter time was obtained using an optimizing compiler.

The analysis in the original proposal used a range of 3 to 45 minutes for 10<sup>6</sup> points to be transformed. If rigorous information is going to be used for each point only the PDP11/45 could not perform adequately.

$10^4$  512 byte blocks. Table 5 contains the results of 4 tests of  $10^4$  sequential reads of 512 contiguous blocks. Projecting the timing values obtained into those possible with the specified high speed disk demonstrate that the I/O operations can be performed with sufficient speed, especially if there is concurrent input and output.

The benchmark tests have shown that current minicomputers do have the power and speed to compete with large main frames such as the UNIVAC 1100/81. The high reliability and low cost of minicomputers strongly suggest that they should be considered not only for the implementation of the algorithm in this report, but also for other algorithms performed by the MARS system. The following section gives recommendations on the implementation of this and other algorithms on both minicomputers and specialized hardware.

Table 4.  
Random Reads from Disk

10000 random reads of 512 byte blocks  
RK equivalent disk with 70 msec avg. access  
PDP 11/45 under UNI X

Test #	# Users	wall clock time	user mode time	system mode time
1	>3	13 min 50 sec	4 sec	69 sec
2	1	11 min 30 sec	4 sec	68 sec
3	1	11 min 30 sec	4 sec	69 sec
4	1	11 min 29 sec	4 sec	66 sec

\* each of 4 tests read same pseudo-random sequence of 10000 blocks

Table 5.  
Sequential Reads from Disk

10000 reads of contiguous 512 byte blocks, system environment as above

Test #	# Users	wall clock time	user mode time	system mode time
1	1	2 min 19 sec	1 sec	31 sec
2	1	3 min 4 sec	1 sec	31 sec
3	1	2 min 43 sec	1 sec	30 sec
4	1	2 min 41 sec	1 sec	30 sec

## 5. CONCLUSIONS AND RECOMMENDATIONS

### 5.1 General Conclusions

(G1) The MARS regridding step can be performed in a very simple manner as originally proposed, at a rate of  $10^6$  points in 40 to 80 minutes, and on hardware costing roughly \$125,000.

This conclusion is based on the decomposition of the processing into simple steps each one of which is tractable to analysis and can be calibrated by benchmarks. Actual benchmarks for I/O and computation, reported in Section 4, support the previous conclusion.

(G2) The current MARS software system could be transported to a minicomputer system with no loss in execution speed and a great saving in execution cost.

The minicomputers studied can have large memories and fast arithmetic and could easily host the current MARS. Throughput should increase if the system were dedicated to MARS through decreased overhead and I/O contention. Moreover, memory limitations are known to slow down the current UNIVAC MARS. Large memory increases possible in the mini systems could speed up the processing by reducing fragmentation. Costs of converting the current UNIVAC system would be high because of the complexity of that system - probably higher than the cost of creating the proposed Mini-MARS software system. Conversion of the current MARS was not actually a task of the current project, however, given the current information it is easy to evaluate such an option.

## 5.2 Conclusions Regarding Coordinate Transformations

(T1) Rigorous transformation of every grid point can double execution time relative to regriding with a large amount of interpolation.

This conclusion is based on data from the MARS system contained in the work proposal and confirmed by subsequent analysis and benchmarks. Actually, rigorous transformation creates a 50/50 mix of computation and I/O time. Minicomputers with slower arithmetic but comparably fast I/O would cause a strong shift toward being compute bound. See Appendix A for a study of the MARS transformation routines.

(T2) Rigorous transformation should be avoided on all systems considered.

From T1 it is seen that even on very fast conventional computers (i.e. UNIVAC 1108, PERKIN-ELMER 3240) half of the process time will be taken up in transformation. Of the computers studied only the UNIVAC could meet the 80 minute process time if all points were regrided rigorously.

The PDP 11's would require several hours for rigorous transformation but could finish in less than 80 minutes using interpolation.

(T3) Provided that only 1 of 200 grid points were transformed rigorously all of the computers studied (PDP 11, S/250, UNIVAC 1108, 3220 and 3240) could regrid  $10^6$  points in less than 80 minutes.

This is a direct consequence of the analysis of the algorithms and the timings from Tables 2 and 3. Past experience with MARS indicates that the fraction 1 of 200 is very reasonable; we will seldom have to rigorously transform more points than this.

(T4) At this point in time special off-the-shelf hardware to speed up the transformations does not appear to be cost effective.

The DG array processor, in the \$14,000 range, is the only inexpensive array processor available for any of the studied machines but it does not handle double precision and is not a true parallel processor. A recently announced Floating Point 54-bit array processor is in the \$150,000 - \$200,000 range. Since such a device could only half the computation time, for example on the PE3240, it would be wiser to double capacity by having two conventional systems. Future reductions in cost and interfacing problems would cause us to reconsider.

(T5) The costs of the 16 bit and 32 bit machines are close enough together that the selection of the 32 bit machine appears to be best. (This machine can be either the 3220 or 3240, or one not reviewed by L.N.K.).

The capability for future expansion and the easy addressing methods of the 32 bit CPU are very strong assets.

### 5.3 Final Conclusions

The Mini-MARS project can be implemented on a mini-computer. It is recommended that a 32 bit machine be selected. The MARS system even in its present form could be implemented on the same machine (as it now exists) with some increase in thru-put. Hardware costs will be in the proposed range of \$100K to \$200K. Setup and software costs should be within \$100K to \$200K.

APPENDIX A

## STUDY OF DMA TRANSFORMATION PACKAGE

The DMA coordinate transformation package contains several superb algorithms for performing coordinate conversion. Two subroutines, however, require modification in order to avoid possible incorrect computation. The suggested changes are given in Section A.

If a large set of points (10,000 points, for instance) are to be processed using the subroutines, some reduction in computation time can be achieved by avoiding recomputing constants such as  $\pi/2$ , the eccentricity of the spheroid, coefficients for equations, etc., in each call of a subroutine. The number of exponentiations, multiplications and divisions should be minimized. Specific suggested modifications in this direction are given in Section B. It should be pointed out that the saving in computation time using such modification is not likely to exceed 10%. On the otherhand, if such economization is combined with improvements in the computation of the trigonometric functions, the total saving in computing time can exceed 25%. In Section C, methods for reducing the computation time of the trigonometric functions are discussed. The actual FORTRAN package is given at the end of this memo.

### A. Corrections and Safeguard Steps

1. FLHUPS, line 9: Change "CON = 1.0" to "CON = 2.0D0".

The derivation of the value for CON can be found in Adler ( [1] ), page 97.

2. FLHUPS: Change all occurrences of the FORTRAN library functions ABS, SIN, COS and SQRT, to DABS, DSIN, DCOS AND DSQRT, respectively. This is a precautionary step.
3. FLHXYZ, line 3: Change "C THIS PROGRAM TRANSFORMS GEOGRAPHIC COORDINATES TO LOCAL" to "C THIS PROGRAM TRANSFORMS GEOGRAPHIC COORDINATES TO GEOCENTRIC".

4. UPSFLH, line 10: Change "CON = 1.0" to "CON = 2.000".
5. UPSFLH: Change all occurrences of the FORTRAN library functions ATAN, ABS, SIN, COS and SQRT to DATAN, DABS, DSIN, DCOS and DSQRT, respectively.
6. XYZFLH, line 28: The variable "A" should be assigned the correct value before this step is executed. Such an assignment step seems to be missing. Also the logic in the expression itself

$$FLH(3) = A - ABS(Z)$$

appears to contradict the definition of geographic coordinates (unless this is a special convention used at DMA). The formula one would generally use is

$$FLH(3) = DABS(Z) - AXES(2).$$

7. UTMFLH: The Newton iteration loop (lines 33-37) should contain an accuracy check in order to (i) alert the user to possible inaccurate results, and (ii) eliminate unnecessary loops when an accurate result has been obtained. See lines 47-50 of UPSFLH for an example of such a checking device.

B. Reducing Computation Time (Economization of Arithmetic Operations)

1. In an initialization subroutine, initialize PI and PIHALF:

$$PI = 3.141592653589793D0$$

$$PIHALF = PI/2.000$$

Save the results in a COMMON area.

Change every occurrence of "PI/2.0" to "PIHALF":

line 14 of FLHUPS,

line 51 of UPSFLH,

lines 21, 25, 27, 30 of XYZFLH.

2. In an initialization subroutine, compute the eccentricity E and its square E2:

C A is semi-major axis and B is semi-minor axis

A = .....

B = .....

$E2 = (A + B) * (A - B) / A / A$

E = DSQRT(E2)

Save the values of E and E2 in a COMMON area.

Remove lines 12, 13 of FLHUPS,  
line 30 of FLHUTM,  
lines 12, 13 of FLHXYZ,  
lines 25, 26 of UPSFLH,  
line 20 of UTMFLH,  
line 41 of XYZFLH.

3. In an initialization subroutine, compute the coefficients A, B, C and D (lines 32-35 of FLHUTM). Save the results in a COMMON area. E and E2 should be computed before computing A, B, C and D.

4. Change lines 19-21 of FLHUPS to

$TEMPL = (1.000 - E) / (1.000 + E)$

$R = CON * (A/B) * TANZ * A * (TEMP * TEMPL) ** EX.$

Change line 43 of UPSFLH to

$$\text{TEMP2} = \text{TEMP1}/\text{TEMP}.$$

In each case, an exponentiation operation is eliminated. Exponentiation is costly: approximately eight multiplications are necessary to perform an exponentiation operation.

5. In UPSFLH, replace lines 34 and 35 with

$$\begin{aligned} \text{S2} &= 2.0\text{D0} * \text{S1} * \text{C1} \\ \text{C2} &= 2.0\text{D0} * \text{C1} * \text{C1} - 1.0\text{D0}. \end{aligned}$$

This modification eliminates one call of the function DSIN(P) and one call of the function DCOS(P).

6. In XYZFLH, add the line "RXY = X\*X + Y\*Y" after the line 42.

Change line 45 to "R = RXY + ZP\*ZP", and change line 54 to "R = RXY".

### C. Computation of Trigonometric Functions

In many subroutines, the time required to compute the trigonometric functions is significant. As a measure of this segment of computing time, we estimate the ratio R:

$$R = (\text{Total number of multiplications and divisions necessary to compute the trigonometric functions in subroutine}) / (\text{Total number of multiplications and divisions performed in each call of the subroutine})$$

If a DO-loop is to be executed in the subroutine, the ratio R takes into account only the operation counts involved in one loop. The ratio will decrease somewhat if more than one loop is considered. The economization modifications of Section B are assumed.

Fortran trigonometric functions are usually implemented using Chebyshev polynomials or Lagrange interpolation polynomials. An example of an approximating polynomial for  $\cos \frac{1}{4}\pi x$  is

$$.9999999724 - .3084242527 x^2 + .0158499130 x^4 - .0003188790 x^6,$$

Fike ( [2] ), page 133. In this case, four multiplications are performed during the evaluation of the polynomial using the factored form  $a + ( (b + (c + dx^2)x^2 )x^2$  with a, b, c, d as above.

<u>SUBROUTINE</u>	<u>R (approximate)</u>
FLHUPS -----	50%
FLHUTM -----	33%
FLHXYZ -----	50%
LCGO -----	0%
(LCGO calls FLHXYZ and ORTM6A)	
MOVE -----	0%
(MOVE is called by TRANCD only)	
TRANCD -----	0%
(TRANCD is an administrative subroutine. It calls other subroutines.)	
UPSFLH -----	35%
UTMFLH -----	40%
XYZFLH -----	20%

The above ratios justify considering "hard-wiring" the trigonometric functions, implementing faster piecewise polynomial approximations.

Note the following computation times cited for DSIN along with the source.

25 x 10<sup>-6</sup> second      DG S/250 Eclipse with microcoded functions. Timing gotten from technical literature.

40 x 10<sup>-6</sup> second      Univac 1108. Timing gotten from actual benchmark program run.

97 x 10<sup>-6</sup> second      Perkin Elmer 3220. Timing gotten from technical literature.

Benchmarks actually run with 10<sup>4</sup> points transformed from geographic to UTM coordinates are as follows.

7.1 sec	Univac 1108
23.0 sec	Perkin Elmer 3220
12.8 sec	Perkin Elmer 3240

D. Documentation

Specific source-references should be given for the approximation formulas in FLHUTM and UTMFLH, since the formulas in the two subroutines are modifications of the original defining equations of the UTM coordinates.

#### REFERENCES

- [1] Adler, R.K., Richardus, P., Map Projections for Geodesists, Cartographers and Geographers, North-Holland Publishing Company - Amsterdam, American Elsevier Publishing Company, Inc., - U.S.A. and Canada, 1972.
- [2] Fike, C.T., Computer Evaluation of Mathematical Functions, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1968.

## DMA TRANSFORMATION SOFTWARE

Following is a listing of all transformation software currently in use by DMA. Line numbers in these listings are those referenced in the prior text. Four driver programs are listed before the subroutines in the package. These drivers, perhaps with some changes in constants, were used to run benchmarks on the transformation software. Benchmarks were run on two different Univac installations (1110 and 1180) and two different Perkin Elmer installations (3220 and 3240).

```

000001      000  C DRIVER TO CALL TRANCD FOR GEOGRAPHIC TO UTM CONVERSION
000002      000  C THIS DRIVER SHORT CUTS BY NOT CALLING ROUTINE TRANCD*****
000003      000  C ALSO WRITTEN 'UNIVAC FREE' TO GO TO PERKIN-ELMER FOR BENCHMARK
000004      000  C
000005      000          IMPLICIT INTEGER(A-Z)
000006      000          DOUBLE PRECISION PTIN(3),PTOUT(3),DAXES(2),PO(3),HO,FLAG,ZONE,DPI
000007      000          DOUBLE PRECISION F(4)
000008      000  C
000009      000          KOUNT = 0
000010      000          DPI = 3.14159265358979324D0
000011      000          PTIN(1) = 10.*DPI/180.
000012      000          PTIN(2) = 10.*DPI/180.
000013      000          PTIN(3) = 0.0D0
000014      000          DAXES(1) = 6378135D0
000015      000          DAXES(2) = 6356750.52D0
000016      000  C CODE ADDED TO SHUNT CALL TO TRANCD
000017      000          F(3)=DAXES(1)
000018      000          F(4)=DAXES(2)
000019      000          PO(1) = 10.0D0
000020      000          PO(2) = 20.*DPI/180.
000021      000          PO(3) = 100.*DPI/180.
000022      000          HO = 0.0D0
000023      000          FLAG = 0.0D0
000024      000          ZONE = 0.0D0
000025      000          KFLAG = 7
000026      000          DO 50 I = 1,200
000027      000          DO 50 J=1,50
000028      000          PTIN(1) = PTIN(1)+1.*DPI/180.
000029      000          PTIN(2) = PTIN(2)+1.*DPI/180.
000030      000          PTIN(3) = PTIN(3)+1.
000031      000          KOUNT = KOUNT + 1
000032      000  C          CALL TRANCD(PTIN,PTOUT,DAXES,PO,HO,FLAG,ZONE,KFLAG)
000033      000  C HERE IS CODE DONE IN TRANCD ENABLING THAT ROUTINE TO BE SHUNTED
000034      000          F(1)=PTIN(1)
000035      000          F(2)=PTIN(2)
000036      000          CALL FLHUTH(F,PTOUT,ZONE)
000037      000          PTOUT(3)=PTIN(3)
000038      000          IF (I.EQ.1.AND.J.EQ.1) WRITE(6,49) PTIN,PTOUT
000039      000          IF(I.EQ.200.AND.J.EQ.50) WRITE(6,49) PTIN,PTOUT
000040      000  49  FORMAT(' PTIN ',3D20.10,/, ' PTOUT ',3D20.10,/)
000041      000  50  CONTINUE
000042      000          WRITE(6,100)KOUNT
000043      000  100  FORMAT('NORMAL EXIT, COUNT = ',I6)
000044      000          STOP
000045      000          END

```

```

1      C DRIVER TO CALL TRANCD FOR UNAMACE TO GEOGRAPHIC CONVERSION
2      C
3      IMPLICIT INTEGER(A-Z)
4      DOUBLE PRECISION PTIN(3),PTOUT(3),DAXES(2),PO(3),HO,FLAG,ZONE,DPI
5      C
6      KOUNT = 0
7      PTIN(1) = -5000.0D0
8      PTIN(2) = -5000.0D0
9      DAXES(1) = 6378135D0
10     DAXES(2) = 6356750.52D0
11     DPI = 3.14159265358979324D0
12     PO(1) = 0.0      @LATITUDE
13     PO(2) = 20.*DPI/180.  @LONGITUDE
14     PO(3) = 100.*DPI/180. @AZIMUTH
15     HO = 0.0D0
16     FLAG = 0.0D0
17     ZONE = 0.0D0
18     KFLAG = 16      @ FOR UNA TO GEO
19     DO 50 I = 1,10
20     PTIN(3) = 0.0D0
21     PO(1) = PO(1)+5.*DPI/180.
22     DO 50 J = 1,10000
23     PTIN(1) = PTIN(1)+1.
24     PTIN(2) = PTIN(2)+1.
25     PTIN(3) = PTIN(3)+1.
26     KOUNT = KOUNT + 1
27     50  CALL TRANCD(PTIN,PTOUT,DAXES,PO,HO,FLAG,ZONE,KFLAG)
28     WRITE(6,100)KOUNT
29     100  FORMAT('ONORMAL EXIT, COUNT = ',I6)
30     STOP
31     END

```

```

1 C DRIVER TO CALL TRANCD FOR UNAMACE TO UTM CONVERSION
2 C
3 IMPLICIT INTEGER(A-Z)
4 DOUBLE PRECISION PTIN(3),PTOUT(3),DAXES(2),PO(3),HO,FLAG,ZONE,DPI
5 C
6 KOUNT = 0
7 PTIN(1) = -5000.0D0
8 PTIN(2) = -5000.0D0
9 DAXES(1) = 6378135D0
10 DAXES(2) = 6356750.52D0
11 DPI = 3.14159265358979324D0
12 PO(1) = 0.0 @LATITUDE
13 PO(2) = 20.*DPI/180. @LONGITUDE
14 PO(3) = 100.*DPI/180. @AZIMUTH
15 HO = 0.0D0
16 FLAG = 0.0D0
17 ZONE = 0.0D0
18 KFLAG = 17 @ FOR UNA TO UTM
19 DO 50 I = 1,10
20 PTIN(3) = 0.0D0
21 PO(1) = PO(1)+5.*DPI/180.
22 DO 50 J = 1,10000
23 PTIN(1) = PTIN(1)+1.
24 PTIN(2) = PTIN(2)+1.
25 PTIN(3) = PTIN(3)+1.
26 KOUNT = KOUNT + 1
27 50 CALL TRANCD(PTIN,PTOUT,DAXES,PO,HO,FLAG,ZONE,KFLAG)
28 WRITE(6,100)KOUNT
29 100 FORMAT('ONORMAL EXIT, COUNT = ',I6)
30 STOP
31 END

```

```

1 C DRIVER TO CALL TRANCD FOR GEOGRAPHIC TO UTM CONVERSION
2 C
3 IMPLICIT INTEGER(A-Z)
4 DOUBLE PRECISION PTIN(3),PTOUT(3),DAXES(2),PO(3),HO,FLAG,ZONE,DPI
5 C
6 KOUNT = 0
7 PTIN(1) = 10.*DPI/180.
8 PTIN(2) = 10.*DPI/180.
9 PTIN(3) = 0.0D0
10 DAXES(1) = 6378135D0
11 DAXES(2) = 6356750.52D0
12 DPI = 3.14159265358979324D0
13 PO(1) = 10.0D0 @LATITUDE
14 PO(2) = 20.*DPI/180. @LONGITUDE
15 PO(3) = 100.*DPI/180. @AZIMUTH
16 HO = 0.0D0
17 FLAG = 0.0D0
18 ZONE = 0.0D0
19 KFLAG = 7 @ FOR GEO TO UTM
20 DO 50 I = 1,2000
21 DO 50 J = 1,50
22 PTIN(1) = PTIN(1)+1.*DPI/180.
23 PTIN(2) = PTIN(2)+1.*DPI/180.
24 PTIN(3) = PTIN(3)+1.
25 KOUNT = KOUNT + 1
26 50 CALL TRANCD(PTIN,PTOUT,DAXES,PO,HO,FLAG,ZONE,KFLAG)
27 WRITE(6,100)KOUNT
28 100 FORMAT('ONORMAL EXIT, COUNT = ',I6)
29 STOP
30 END

```

1	SUBROUTINE FLHUPS (PTIN,PTOUT)	FLH00100
2	C	FLH00200
3	C THIS ROUTINE GEOGRAPHIC COORDINATES TO UNIVERSAL POLAR STEREOGRAPHIC	FLH00300
4	C	FLH00400
5	IMPLICIT DOUBLE PRECISION (A-H,O-Z)	FLH00500
6	DIMENSION PTIN(5),PTOUT(2)	FLH00600
7	C	FLH00700
8	PI = 3.141592653589793	FLH00800
9	CON = 1.0	FLH00900
10	A = PTIN(4)	FLH01000
11	B = PTIN(5)	FLH01100
12	E = 1. - (B/A)**2	FLH01200
13	E = SQRT (E)	FLH01300
14	P = PI/2.0 - ABS (PTIN(1))	FLH01400
15	TANZ = SIN (P/2.0) / COS (P/2.0)	FLH01500
16	COSP = COS (P)	FLH01600
17	TEMP = (1. + E * COSP) / (1. - E * COSP)	FLH01700
18	EX = E/2.	FLH01800
19	TANZ = TANZ * TEMP**EX	FLH01900
20	TEMP = (1. - E) / (1. + E)	FLH02000
21	R = CON * (A/B) * TANZ * A * TEMP**EX	FLH02100
22	PTOUT(1) = R * COS (PTIN(2)) + 2000000.0	FLH02200
23	Y = R * SIN (PTIN(2))	FLH02300
24	IF (PTIN(1)) 10,20,20	FLH02400
25	10 PTOUT(2) = 2000000.0 + Y	FLH02500
26	GO TO 30	FLH02600
27	20 PTOUT(2) = 2000000.0 - Y	FLH02700
28	30 RETURN	FLH02800
29	END	FLH02900

FLHUTM 1 of 2

```
1      SUBROUTINE FLHUTM(F,O,ZONE)
2      C      GEOGRAPHIC TO UTM
3      C      IMPLICIT DOUBLE PRECISION (A-H,L-Z)
4      C      DIMENSION F(4), O(2)
5      C      F AND O ARE THE SAME CALLING VARIABLES THAT APPEAR IN GENCOR
6      C      F(1) (OR L1) IS THE LATITUDE IN RADIANS
7      C      F(2) (OR L2) IS THE LONGITUDE IN RADIANS
8      C      F(3) (OR A1) AND F(4) (OR B1) ARE THE AXES OF THE SPHEROID.  A1 GT B1.
9      C      L1=F(1)
10     C      L2=F(2)
11     C      A1=F(3)
12     C      B1=F(4)
13     C      Z = ZONE
14     C      Y IS A CONVERSION FACTOR DEG TO RAD
15     C      Y=.0174532925199432958D0
16     C      IF (ZONE.GT.60.0D0) ZONE = 0.0D0
17     C      LONG = L2/Y
18     C      IF (LONG.LT.-180.0D0) LONG = LONG + 360.0D0
19     C      IF (LONG.GT.180.0D0) LONG = LONG - 360.0D0
20     C      L2 = LONG * Y
21     C      IF (ZONE.LE.0.0D0) GO TO 1      @ZONE NOT YET COMPUTED
22     C
23     C      CHECK FOR ZONE OVERLAP ON EITHER SIDE OF 180/-180 DEGREES
24     C      LONGITUDE. FORCE CONSISTENT LONGITUDE VALUE.
25     C
26     C      IF (LONG.LT.0.0D0.AND.ZONE.EQ.60.0D0) LONG = LONG + 360.0D0
27     C      IF (LONG.GT.0.0D0.AND.ZONE.EQ.1.0D0) LONG = LONG - 360.0D0
28     C      L2 = LONG * Y      @RECOMPUTE L2 FOR OVERLAP
29     C      E IS THE ECCENTRICITY SQUARED
30     C      E=(A1-B1)*(A1+B1)/A1/A1
31     C      A, B, C, D DEPEND ONLY ON E AND ARE AS IN THE WRITE UP
32     C      A=1.+E*(.75+E*(0.703125D0+E*.68359375D0))
33     C      B=E*(0.375+E*(0.46875+E*0.512953125D0))
34     C      C=E*E*(.05859375D0+E*.1025390625D0)
35     C      D=E*E*E*0.01139322916666667D0
36     C      C1=DCOS(L1)
37     C      S1=DSIN(L1)
38     C      T=DTAN(L1)
39     C      S2=C1*C1*E/(1.-E)
40     C      T2=T*T
```

```

41 C N IS THE RADIUS OF CURVATURE IN THE PRIME VERTICAL
42 N=A1/DSQRT(1.-E*S1*S1)
43 C3=1.-T2+S2
44 C4=T*(5.-T2+S2*(9.+4.*S2))
45 C5=5.+T2*(T2-18.)+S2*(14.-58.*T2)
46 C6=T*(61.+T2*(T2-58.)+S2*(270.-330.*T2))
47 IF (ZONE.GT.0.0D0) GO TO 3 @FORCE PREVIOUS ZONE
48 IF (L2.GE.0.0) GO TO 2
49 C Z IS THE ZONE
50 Z=30.+DINT(LONG/6.)
51 IF (Z.EQ.0.0D0) Z = 60.0D0 @FOR -180 DEGREES LONGITUDE
52 GO TO 3
53 2 Z=31.+DINT(LONG/6.)
54 C M IS THE CENTRAL MERIDIAN IN RADIANS
55 3 M=(6.*Z-183.)*Y
56 C S IS THE ARC LENGTH ALONG THE CENTRAL MERIDIAN
57 S=A1*(1.-E)*(A*L1-B*DSIN(2.*L1)+ C*DSIN(4.*L1)-D*DSIN(6.*L1))
58 D2=(L2-M)*C1
59 D3=D2*D2
60 C CHECK TO AVOID UNDERFLOW
61 IF (ABS(D2).LT.1.0E-5) GO TO 10
62 E2= M*D2*(1.+D3*(C3/6.+D3*C5/120.))*9996D0
63 N2=.9996D0*S+.9996D0*N*D3*(.5*T+D3*(C4/24.+D3*C6/720.))
64 GO TO 20
65 10 E2= M*D2*(1.+D3*C3/6.)*9996D0
66 N2=.9996D0*S+.9996D0*N*D3*T*.5
67 20 E2=E2+5.D5
68 IF(N2.GE.0.) GO TO 30
69 N2=N2+1.D7
70 C O(1) (OR E2) IS THE UTM EASTING
71 C O(2) (OR N2) IS THE UTM NORTHING
72 30 O(1)=E2
73 O(2)=N2
74 IF (ZONE.GT.0.0D0) GO TO 899
75 ZONE = 2
76 899 RETURN
77 END

```

1		SUBROUTINE FLHXYZ(FI,X,AXES)	FLH00100
2	C		FLH00200
3	C	THIS PROGRAM TRANSFORMS GEOGRAPHIC COORDINATES TO LOCAL	FLH00300
4	C		FLH00400
5		IMPLICIT DOUBLE PRECISION (A-H,O-Z)	FLH00500
6		DIMENSION FI(3),X(3),AXES(2)	FLH00600
7	C		FLH00700
8		SINFI = DSIN (FI(1))	FLH00
9		COSFI = DCOS (FI(1))	FLH00
10		SINL = DSIN (FI(2))	FLH00
11		COSL = DCOS (FI(2))	FLH00
12		ALFA = (AXES(2)/AXES(1))*2	FLH01200
13		E2 = 1. - ALFA	FLH01300
14		GAMA = 1. - E2*(SINFI**2)	FLH01400
15		GAMA = DSQRT (GAMA)	FLH0
16		FN = AXES(1)/GAMA	FLH01600
17		GAMA = (FN+FI(3)) * COSFI	FLH01700
18		X(1) = GAMA * COSL	FLH01800
19		X(2) = GAMA * SINL	FLH01900
20		X(3) = (FN * ALFA + FI(3)) * SINFI	FLH02000
21		RETURN	FLH02100
22		END	FLH02200

1		SUBROUTINE LCGO(PIN,POUT,AXES,ORIG,HO,MODE)	LCG00100
2	C		LCG00200
3	C	THIS PROGRAM TRANSFORMS COORDINATES FROM LOCAL TO GEOCENTRIC AND VICE	LCG00300
4	C	VERSA	LCG00400
5	C		LCG00500
6	C	MODE = 1 , GEOCENTRIC TO LOCAL	LCG00600
7	C	MODE = 2 , LOCAL TO GEOCENTRIC	LCG00700
8	C		LCG00800
9		IMPLICIT DOUBLE PRECISION (A-H,O-Z)	LCG00900
10		DIMENSION PIN(3),POUT(3),AXES(2),ORIG(3),RT(3,3),PO(3)	LCG01000
11	C		LCG01100
12		POUT(1) = ORIG(1)	LCG01200
13		POUT(2) = ORIG(2)	LCG01300
14		POUT(3) = HO	LCG01400
15		CALL FLHXYZ(POUT,PO,AXES)	LCG01500
16		CALL ORTM6A(ORIG,RT)	LCG01600
17		GO TO (10,30),MODE	LCG01700
18	10	DO 20 I = 1,3	LCG01800
19		POUT(I) = 0.0	LCG01900
20		DO 20 J = 1,3	LCG02000
21		POUT(I) = POUT(I)+RT(I,J)*(PIN(J)-PO(J))	LCG02100
22	20	CONTINUE	LCG02200
23		GO TO 50	LCG02300
24	30	DO 40 I = 1,3	LCG02400
25		POUT(I) = PO(I)	LCG02500
26		DO 40 J = 1,3	LCG02600
27		POUT(I) = POUT(I)+RT(J,I)*PIN(J)	LCG02700
28	40	CONTINUE	LCG02800
29	50	RETURN	LCG02900
30		END	LCG03000

1		SUBROUTINE MOVE(A,B,N)	MOV00100
2	C		MOV00200
3	C		MOV00300
4		IMPLICIT DOUBLE PRECISION (A-H,O-Z)	MOV00400
5	C		MOV00500
6	C		MOV00600
7	C		MOV00700
8	C		MOV00800
9		DIMENSION A(1),B(1)	MOV00900
10		DO 10 I = 1,N	MOV01000
11		10 B(I) = A(I)	MOV01100
12		RETURN	MOV01200
13		END	MOV01300

1		SUBROUTINE ORTM6A(V,A)	DRT00100
2			DRT00200
3		IMPLICIT DOUBLE PRECISION (A-H,O-Z)	DRT00300
4		DIMENSION V(3),A(3,3),C(3),S(3)	DRT00400
5	C		DRT00500
6		DO 10 I = 1,3	DRT00600
7		C(I) = COS (V(I))	DRT00700
8		10 S(I) = SIN (V(I))	DRT00800
9		A(1,1) = S(3)*S(1)*C(2) - S(2)*C(3)	DRT00900
10		A(1,2) = S(3)*S(1)*S(2) + C(2)*C(3)	DRT01000
11		A(1,3) = -S(3)*C(1)	DRT01100
12		A(2,1) = -C(3)*S(1)*C(2)-S(2)*S(3)	DRT01200
13		A(2,2) = -C(3)*S(1)*S(2) + C(2)*S(3)	DRT01300
14		A(2,3) = C(3)*C(1)	DRT01400
15		A(3,1) = C(1)*C(2)	DRT01500
16		A(3,2) = C(1)*S(2)	DRT01600
17		A(3,3) = S(1)	DRT01700
18		RETURN	DRT01800
19		END	DRT01900

1		SUBROUTINE TRANCD (R,S,SEMI,P,HO,FLAG,ZONE,KFLAG)	TRA00100
2	C		TRA00200
3		IMPLICIT DOUBLE PRECISION (A-H,O-Z)	TRA00300
4		DIMENSION R(3),S(3),P(3),F(6),SEMI(2),RP(3)	TRA00400
5	C		TRA00500
6	C	KFLAG DETERMINES THE FORM OF THE OUTPUT AND INDICATES THE FORM OF	TRA00600
7	C	THE INPUT. THE POSSIBILITIES ARE AS FOLLOWS	TRA00700
8	C	KFLAG=1 MEANS UTM TO GEOGRAPHIC	TRA00800
9	C	=2 MEANS UTM TO GEOCENTRIC	TRA00900
10	C	=3 MEANS UTM TO LOCAL RECTANGULAR	TRA01000
11	C	=4 MEANS UPS TO GEOGRAPHIC	TRA01100
12	C	=5 MEANS UPS TO GEOCENTRIC	TRA01200
13	C	=6 MEANS UPS TO LOCAL RECTANGULAR	TRA01300
14	C	=7 MEANS GEOGRAPHIC TO UTM	TRA01400
15	C	=8 MEANS GEOGRAPHIC TO UPS	TRA01500
16	C	=9 MEANS GEOGRAPHIC TO GEOCENTRIC	TRA01600
17	C	=10 MEANS GEOGRAPHIC TO LOCAL RECTANGULAR	TRA01700
18	C	=11 MEANS GEOCENTRIC TO UTM	TRA01800
19	C	=12 MEANS GEOCENTRIC TO UPS	TRA01900
20	C	=13 MEANS GEOCENTRIC TO GEOGRAPHIC	TRA02000
21	C	=14 MEANS GEOCENTRIC TO LOCAL RECTANGULAR	TRA02100
22	C	=15 MEANS LOCAL RECTANGULAR TO GEOCENTRIC	TRA02200
23	C	=16 MEANS LOCAL RECTANGULAR TO GEOGRAPHIC	TRA02300
24			
25	C	=17 MEANS LOCAL RECTANGULAR TO UTM	TRA02400
26	C	=18 MEANS LOCAL RECTANGULAR TO UPS	TRA02500
27	C		TRA02600
28	C	P(3X1) IS A VECTOR, (LATITUDE, LONGITUDE, AZIMUTH) OF THE LOCAL SYSTEM.	TRA02700
29	C	THE ALTITUDE OF THE LOCAL RECTANGULAR SYSTEM IS ZERO.	TRA02800
30	C	A AND B ARE THE MAJOR AND MINOR SEMIAXES OF THE SPHEROID.	TRA02900
31	C	FLAG = 0. IF IN THE NORTHERN HEMISPHERE, =1 IN THE SOUTHERN	TRA03000
32	C	R AND S ARE THE INPUT AND OUTPUT COORDINATES, RESPECTIVELY	TRA03100
33	C	E X LATITUDE	TRA03200
34	C	S= N OR = Y OR = LONGITUDE SIMILARLY FOR R.	TRA03300
35	C	H Z ALTITUDE	TRA03400
36	C		TRA03500
37		IF (KFLAG) 220,220,10	TRA03600
38		10 IF (KFLAG-18) 20,20,220	TRA03700
39		20 IF (KFLAG-6) 100,100,25	TRA03800
40		25 FLAG = 0.0	TRA03900
41		IF (KFLAG-7) 30,40,30	TRA04000
42		30 IF (KFLAG-8) 60,40,60	TRA04100
43		40 IF (R(1)) 90,100,100	TRA04200
44		60 IF (KFLAG-11) 70,80,70	TRA04300
45		70 IF (KFLAG-12) 100,80,100	TRA04400
46		80 IF (R(3)) 90,100,100	TRA04500
47		90 FLAG = 1.0	TRA04600
48		100 CALL MOVE (R,RP,3)	TRA04700
49		60 TO(1,1,1,2,2,2,3,4,5,5,6,6,6,7,8,8,8,8),KFLAG	TRA04800
50	C		TRA04900
51	C	UTM TO GEOGRAPHIC	TRA05000
52	C		TRA05100
53		1 F(1) = RP(1)	TRA05200
54		F(2)=RP(2)	TRA05300
55		F(3)=ZONE	TRA05400
56		F(4)=SEMI(1)	TRA05500
57		F(5)=SEMI(2)	TRA05600
58		F(6)=FLAG	TRA05700
59		CALL UTMFLH (F,8)	TRA05800
60		B(3)=RP(3)	TRA05900
61		IF (KFLAG-1) 110,220,110	TRA06000
62		110 CALL MOVE(S,RP,3)	TRA06100

TRANCD

part 1 of 2

63	C		TRA06200
64	C	GEOGRAPHIC TO GEOCENTRIC	TRA06300
65	C		TRA06400
66		5 CALL FLXYZ (RP,S,SEMI)	TRA06500
67		IF (KFLAG-2) 120,220,120	TRA06600
68		120 IF (KFLAG-5) 130,220,130	TRA06700
69		130 IF (KFLAG-9) 140,220,140	TRA06800
70		140 CALL MOVE (S,RP,3)	TRA06900
71	C		TRA07000
72	C	GEOCENTRIC TO LOCAL RECTANGULAR	TRA07100
73	C		TRA07200
74		7 CALL LCGO (RP,S,SEMI,P,HO,1)	TRA07300
75		GO TO 220	TRA07400
76	C		TRA07500
77	C	UPS TO GEOGRAPHIC	TRA07600
78	C		TRA07700
79		2 F(1)=RP(1)	TRA07800
80		F(2)=RP(2)	TRA07900
81		F(3) = FLAG	TRA08000
82		F(4) = SEMI(1)	TRA08100
83		F(5) = SEMI(2)	TRA08200
84		CALL UPSFLH (F,S)	TRA08300
85		S(3)=RP(3)	TRA08400
86		IF (KFLAG-4) 110,220,110	TRA08500
87	C		TRA08600
88	C	GEOGRAPHIC TO UTM	TRA08700
89	C		TRA08800
90		3 F(1)=RP(1)	TRA08900
91		F(2)=RP(2)	TRA09000
92		F(3) = SEMI(1)	TRA09100
93		F(4) = SEMI(2)	TRA09200
94		CALL FLHUTH (F,S,ZONE)	TRA09300
95		S(3)=RP(3)	TRA09400
96		GO TO 220	TRA09500
97	C		TRA09600
98	C	GEOGRAPHIC TO UPS	TRA09700
99	C		TRA09800
100		4 F(1)=RP(1)	TRA09900
101		F(2)=RP(2)	TRA10000
102		F(3) = FLAG	TRA10100
103		F(4) = SEMI(1)	TRA10200
104		F(5) = SEMI(2)	TRA10300
105		CALL FLHUPS (F,S)	TRA10400
106		S(3)=RP(3)	TRA10500
107		GO TO 220	TRA10600
108	C		TRA10700
109	C	GEOCENTRIC TO GEOGRAPHIC	TRA10800
110	C		TRA10900
111		6 CALL XYZFLH(RP,S,SEMI)	TRA11000
112		IF (KFLAG-13) 170,220,170	TRA11100
113		170 IF (KFLAG-16) 180,220,180	TRA11200
114		180 CALL MOVE (S,RP,3)	TRA11300
115		IF (KFLAG-11) 190,3,190	TRA11400
116		190 IF (KFLAG-12) 200,4,200	TRA11500
117		200 IF (KFLAG-17) 4,3,4	TRA11600
118	C		TRA11700
119	C	LOCAL RECTANGULAR TO GEOCENTRIC	TRA11800
120	C		TRA11900
121		8 CALL LCGO (RP,S,SEMI,P,HO,2)	TRA12000
122		CALL MOVE (S,RP,3)	TRA12100
123		IF (KFLAG-15) 6,220,6	TRA12200
124	220	RETURN	TRA12300
125		END	TRA12400

TRANCD

part 2 of 2

1	SUBROUTINE UPSFLH(PTIN,PTOUT)	UPS00100
2	C	UPS00200
3	C THIS SUBROUTINE TRANSFORMS UNIVERSAL POLAR STEREOGRAPHIC COORDINATES	UPS00300
4	C TO GEOGRAPHIC.	UPS00400
5	C	UPS00500
6	IMPLICIT DOUBLE PRECISION (A-H,O-Z)	UPS00600
7	DIMENSION PTIN(5),PTOUT(2)	UPS00700
8	C	UPS00800
9	PI = 3.141592653589793	UPS00900
10	CON = 1.	UPS01000
11	IF(PTIN(3)) 20,10,20	UPS01100
12	10 Y = 2000000. - PTIN(2)	UPS01200
13	FACT = 1.0	UPS01300
14	GO TO 30	UPS01400
15	20 Y = PTIN(2) - 2000000.	UPS01500
16	FACT = -1.0	UPS01600
17	30 X = PTIN(1) - 2000000.	UPS01700
18	PTOUT(2) = ATAN2(Y,X)	UPS01800
19	IF (ABS (X) - ABS (Y)) 40,40,50	UPS01900
20	40 R = Y / SIN (PTOUT(2))	UPS02000
21	GO TO 60	UPS02100
22	50 R = X / COS (PTOUT(2))	UPS02200
23	60 A = PTIN(4)	UPS02300
24	B = PTIN(5)	UPS02400
25	E2 = 1. -(B/A)**2	UPS02500
26	E = SQRT (E2)	UPS02600
27	TEMP = (1. - E) / (1. + E)	UPS02700
28	EX = E/2.	UPS02800
29	TANZ = (R / A) * (B/A) / ((TEMP**EX) * CON)	UPS02900
30	P = ATAN (TANZ) * 2.0	UPS03000
31	DO 70 I = 1,10	UPS03100
32	S1 = SIN (P/2.0)	UPS03200
33	C1 = COS (P/2.0)	UPS03300
34	S2 = SIN (P)	UPS03400
35	C2 = COS (P)	UPS03500
36	TEMP = (1. + E * C2) / (1. - E*C2)	UPS03600
37	TAMP = S1 / C1	UPS03700
38	C = TAMP *(TEMP**EX) - TANZ	UPS03800
39	C12 = C1**2	UPS03900
40	CONST = (1. - E * C2) **2	UPS04000
41	TEMP1 = TEMP **EX	UPS04100
42	EXX = EX - 1.	UPS04200
43	TEMP2 = TEMP **EXX	UPS04300
44	CONST = (.5 * TEMP1/C12) - (E2 * TEMP2 * TAMP * S2 / CONST)	UPS04400
45	DP = - C / CONST	UPS04500
46	P = P + DP	UPS04600
47	IF (ABS (DP) - .000000005) 80,80,70	UPS04700
48	70 CONTINUE	UPS04800
49	PRINT 1000	UPS04900
50	1000 FORMAT (14H2 ERROR UPSFLH)	UPS05000
51	80 PTOUT(1) = (PI/2. - P) * FACT	UPS05100
52	RETURN	UPS05200
53	END	UPS05300

```

1      SUBROUTINE UTMFLH(F,O)
2      C      UTM TO GEOGRAPHIC
3      C      IMPLICIT DOUBLE PRECISION (A-H,L-Z)
4      C      DIMENSION F(6),O(2)
5      C      F AND O ARE THE SAME CALLING VARIABLES THAT APPEAR IN GENCOR
6      C      F(1) (OR E1) IS THE UTM EASTING. F(2) (OR N1) IS THE UTM NORTHING
7      C      F(3) (OR Z) IS THE ZONE
8      C      F(4) (OR A1) AND F(5) (OR B1) ARE THE AXES OF THE SPEROID. A1 GT B1.
9      C      F(6) (OR H) IS 1 OR 0. 1 FOR THE SOUTHERN HEMISPHERE. 0 FOR THE
10     C      NORTHERN HEMISPHERE
11         E1=F(1)
12         N1=F(2)
13         Z=F(3)
14         A1=F(4)
15         B1=F(5)
16         H=F(6)
17     C      Y IS A CONVERSION FACTOR, DEG TO RAD
18         Y=.0174532925199432958D0
19     C      E IS THE ECCENTRICITY SQUARED
20         E=(A1-B1)*(A1+B1)/A1/A1
21     C      A, B, C, D DEPEND ONLY ON E AND ARE AS IN THE WRITE UP
22         A=1.+E*(.75+E*(0.703125D0+E*.68359375D0))
23         B=E*(0.375+E*(0.46875+E*0.512953125D0))
24         C=E*E*(.05859375D0+E*.1025390625D0)
25         D=E*E*E*.0113932291666666667D0
26     C      E2 AND N2 ARE THE UNSCALED UTMS
27         E2=(E1-5.D5)/0.9996D0
28         N2=(N1-H*1.D7)/0.9996D0
29     C      M IS THE CENTRAL MERIDIAN IN RADIANS
30         M=(6.*Z-183.)*Y
31     C      THE NEXT 6 LINES CONTAIN THE NEWTON ITERATION TO COMPUTE THE FOOTPOINT LAT
32         Q=N2/A1/(1.-E)/A
33         DO 10 I=1,5
34             G=N2-A1*(1.-E)*(A*Q-B*DSIN(2.*Q)+C*DSIN(4.*Q)-6.*D*DSIN(6.*Q))
35             G1=-A1*(1.-E)*(A-2.*B*DCOS(2.*Q)+4.*C*DCOS(4.*Q)-6.*D*DCOS(6.*Q))
36             Q=Q-G/G1
37     10    CONTINUE
38     C      THE FINAL VALUE OF Q IS THE FOOTPOINT LATITUDE
39         C1=DCOS(Q)
40         S1=BSIN(Q)
41         T=BTAN(Q)
42         T2=T*T
43     C      M IS THE RADIUS OF CURVATURE IN THE PRIME VERTICAL OF THE FOOTPOINT
44         M=A1/DSQRT(1.-E*S1*S1)
45         S2=C1*C1*E/(1.-E)
46         D2=E2/M
47         D3=D2*D2
48         L2=D2*(1.+D3*(-(1.+2.*T2+S2)/6.+D3*(5.+6.*S2+T2*(28.+8.*S2+24.*T2)
49         2)/120.))/C1*M
50         L1=Q+D3*T*(1.+S2)*(-.5+D3*(5.+S2*(1.-4.*S2)+T2*(3.-9.*S2))/24.)
51     C      O(1) (OR L1) IS THE LATITUDE IN RAD
52     C      O(2) (OR L2) IS THE LONGITUDE IN RAD
53         O(1)=L1
54         LONG = L2/Y
55         IF (LONG.LT.-180.0D0) LONG = LONG + 360.0D0
56         IF (LONG.GT.180.0D0) LONG = LONG - 360.0D0
57         L2 = LONG * Y
58         O(2)=L2
59         RETURN
60         END

```

1	SUBROUTINE XYZFLH(XYZ,FLH,AXES)	XYZ00100
2	C	XYZ00200
3	C THIS SUBROUTINE CONVERTS POINT COORDINATES FROM GEDCENTRIC TO	XYZ00300
4	C GEOGRAPHIC	XYZ00400
5	C THE SUBROUTINE USES DOUBLE PRECISION ARITHMATIC	XYZ00500
6	C	XYZ00600
7	C SPECIFICATIONS	XYZ00700
8	C	XYZ00800
9	IMPLICIT DOUBLE PRECISION (A-H,O-Z)	XYZ00900
10	DIMENSION XYZ(3),FLH(3),AXES(2)	XYZ01000
11	PI =3.141592653589793D0	XYZ010
12	C	XYZ01200
13	X = XYZ(1)	XYZ01300
14	Y = XYZ(2)	XYZ01400
15	Z = XYZ(3)	XYZ01500
16	C	XYZ01600
17	C COMPUTE LONGITUDE LAMDA	XYZ01700
18	C	XYZ01800
19	IF (X) 50,10,60	XYZ01900
20	10 IF (Y) 20,30,40	XYZ02000
21	20 FLH(2) = -PI/2.0	XYZ02100
22	80 TO 80	XYZ02200
23	30 FLH(2) = 0.0	XYZ02300
24	IF (Z) 32,32,33	XYZ02400
25	32 FLH(1) = -PI/2.0	XYZ02500
26	GO TO 34	XYZ02600
27	33 FLH(1) = PI/2.0	XYZ02700
28	34 FLH(3) = A - ABS (Z)	XYZ02800
29	GO TO 120	XYZ02900
30	40 FLH(2) = PI/2.0	XYZ03000
31	GO TO 80	XYZ03100
32	50 COM = -PI	XYZ03200
33	GO TO 70	XYZ03300
34	60 COM = 0.0	XYZ03400
35	70 FLH(2) = DATAN (Y/X) + COM	XYZ00
36	C	XYZ03600
37	C COMPUTE LATITUDE PHI	XYZ03700
38	C	XYZ03800
39	80 A = AXES(1)	XYZ03900
40	B = AXES(2)	XYZ04000
41	E2 = 1. - (B/A)**2	XYZ04100
42	T1 = E2 * Z	XYZ04200
43	DO 100 I = 1,10	XYZ04300
44	ZP = T1 + Z	XYZ04400
45	R = (X**2 + Y**2 + ZP**2)	XYZ04500
46	SINPI = ZP / DSQRT (R)	XYZ00
47	T2 = (A * E2 * SINPI) / DSQRT (1.0 - E2 * SINPI**2)	XYZ00
48	D = DABS (T1 - T2)	XYZ00
49	IF (D-.005) 110,110,90	XYZ04900
50	90 T1 = T2	XYZ05000
51	100 CONTINUE	XYZ05100
52	WRITE (6,1000)	XYZ05200
53	1000 FORMAT (14H2 ERROR XYZFLH)	XYZ05300
54	110 R = X**2 + Y**2	XYZ05400
55	R =DSQRT (R)	XYZ050
56	ZP = Z + T2	XYZ05600
57	FLH(1) = DATAN (ZP/R)	XYZ00
58	DN = A /DSQRT (1. - E2 * SINPI**2)	XYZ050
59	R = ZP / SINPI	XYZ05900
60	FLH(3) = R - DN	XYZ06000
61	120 RETURN	XYZ06100
62	END	XYZ06200

APPENDIX B



Route 4 Westboro, Massachusetts 01581

Telex: 411481 9100 TWX 710 390-0709

QUOTATION

QUOTATION NO  
**181830**

PLEASE REFER TO THIS QUOTATION NO  
IN ALL CORRESPONDENCE AND ORDERS

NEAREST DGC SALES OFFICE

TO: • LNK Corp.  
• 4321 Hartwick Rd.  
• College Park, MD 20740  
•  
• Attn: Russ Smith/Dr. Kanal

• 7927 Jones Branch Dr.  
• McLean, VA 22102  
•  
• 703/827-9600

THANK YOU FOR YOUR INQUIRY. WE ARE PLEASED TO QUOTE AS FOLLOWS:

DATE	REFERENCE	FREIGHT CHARGES	TERMS	NET CASH	FOB POINT OF ORIGIN			
7/29/80		PREPAY AND ADD	Credit Terms on Approval	of D.G.C. Credit Dept				
ITEM	QUANTITY	DESCRIPTION	MTLY PRICE	MAINT TOTAL PRICE	UNIT LIST PRICE	DISC %	UNIT NET PRICE	TOTAL
		<u>DELUXE S250</u> (One disk, two tapes, console)						
1	1	8635-NB S250 with 256KB memory (two-way interleaved)			40,500	15	34,425.00	34,425.00
2	1	8641 High Speed Hardware floating Pt.			6,195	15	5,265.75	5,265.75
3	1	6026 800/1600 Tape w/ Cont.			15,500	15	13,175.00	13,175.00
4	1	6026 800/1600 Add-on Tape Drive			11,300	15	9,605.00	9,605.00
5	1	8650-B Dual Peripheral Bays			2,600	NA	2,600.00	2,600.00
6	1	6067-N 50MB Disk with Cont.			19,800	15	16,830.00	16,830.00
7	1	6040 60 CPS Terminal Printer			2,650	15	2,252.50	2,252.50
		HARDWARE TOTAL-----						84,153.25

ATTACHMENTS:

- DGC Discount Agreement Form \_\_\_\_\_
- DGC Program License Agreement Form 500 \_\_\_\_\_
- DGC Program Availability Schedule Form 501 \_\_\_\_\_
- DGC Maintenance Contract Form \_\_\_\_\_

THIS QUOTATION SHALL REMAIN VALID FOR 30 DAYS FROM THE DATE HEREON. IT IS THE POLICY OF DATA GENERAL CORPORATION (DGC) TO QUOTE ON THE BASIS OF YOUR ORDER. THIS QUOTATION AND ANY ORDER PLACED AS A RESULT HEREOF SHALL BE SUBJECT EXCLUSIVELY TO THE TERMS AND CONDITIONS HEREON AND ON THE REVERSE SIDE AND THE TERMS OF THE ATTACHED AGREEMENTS. ANY CONTRACT RESULTING FROM THIS QUOTATION MUST BE SIGNED IN SOUTHBORO, MASS. BY A DULY AUTHORIZED REPRESENTATIVE OF DGC. ANY ADDITIONS, MODIFICATIONS OR WAIVERS OF ANY OF THE TERMS AND CONDITIONS CONTAINED HEREIN OR ON THE ATTACHED AGREEMENTS SHALL ONLY BE EFFECTIVE IF IN WRITING AND AGREED TO BY AN AUTHORIZED REPRESENTATIVE OF DGC. DGC MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLIED, AS TO THE FITNESS OF THE PRODUCTS QUOTED FOR ANY PARTICULAR PURPOSE, SUCH DETERMINATION TO BE THE SOLE RESPONSIBILITY OF THE CUSTOMER.

tmc

FORM NO. 11000113 04

B-1

BY *Michael Zuckerman*  
Michael Zuckerman, Federal Acct. Mgr.  
CUSTOMER COPY

# DataGeneral

Route 9, Westboro, Massachusetts 01581  
Telephone 617-366-8911

## QUOTATION CONTINUATION SHEET

QUOTATION NO  
181630

PLEASE REFER TO THIS QUOTATION NO  
IN ALL CORRESPONDENCE AND ORDERS

PAGE OF

CPU TYPE		DATE	REFERENCE	FREIGHT CHARGES				
		7/29/80						
ITEM	QUANTITY	DESCRIPTION	MTLY MAINT		UNIT LIST PRICE	DISC %	UNIT NET PRICE	TOTAL
			UNIT PRICE	TOTAL PRICE				
		<u>TO ADD BURST MULTIPLEXOR CHANNEL</u>						
1	1	6067-HN 50MB Disk with Cont (substitute this model for 6067-N on previous page)			21,800	15	18,530.00	18,530.00
2	1	8642 Burst Multiplexor Channel (add this model to previous page)			3,150	15	2,677.50	2,677.50
		REVISED TOTAL-----						88,530.75

\*Model/Feature Defined in Data General Price List

B-2

See Page 1 for Signature

01-2000-1004

CUSTOMER COPY

# DataGeneral

Route 9, Westboro, Massachusetts 01581  
Telephone 617-366-8911

## QUOTATION CONTINUATION SHEET

QUOTATION NO  
**181830**

PLEASE REFER TO THIS QUOTATION NO  
IN ALL CORRESPONDENCE AND ORDERS

PAGE OF

CPU TYPE		DATE	REFERENCE	FREIGHT CHARGES				
		7/29/80						
ITEM	QUANTITY	DESCRIPTION	MTLY MAINT		UNIT LIST PRICE	DISC. %	UNIT NET PRICE	TOTAL
			UNIT PRICE	TOTAL PRICE				
		<u>SUPER DELUXE</u>						
		<u>TO ADD INTEGRAL ARRAY PROCESSOR</u>						
1	1	8652B I/O only module			900	15	765.00	765.00
2	1	8652-C I/O system Module			900	15	765.00	765.00
3	1	8315 I/O Bus Repeater			1,200	15	1,020.00	1,020.00
4	1	8644 Integral Array Processor			14,595	15	12,405.75	12,405.75
5	1	8638 Writeable Control store			4,200	15	3,570.00	3,570.00
REVISED TOTAL-----								107,056.50

\*Model/Feature Defined in Data General Price List

B-3

See Page 1 for Signature

01300011404

CUSTOMER COPY

**DataGeneral**  
 6 Route 9, Westboro, Massachusetts 01581  
 Telephone 617-366-8911

**QUOTATION  
 CONTINUATION SHEET**

QUOTATION NO  
 181830

PLEASE REFER TO THIS QUOTATION NO  
 IN ALL CORRESPONDENCE AND ORDERS

PAGE OF

ITEM	QUANTITY	DESCRIPTION	MTLY MAINT		UNIT LIST PRICE	DISC %	UNIT NET PRICE	TOTAL
			UNIT PRICE	TOTAL PRICE				
		<u>SOFTWARE</u>						
1	1	3359 Real Time Disk Operating System (MRDOS) for Eclipse, Software Subscription Service for one year, one training credit, installation						2,400.00
2	1	3374 Fort. V (Optimizing Compiler), one training credit, installation, software subscription service						2,000.00
3		XXXX Diagnostics						N/C
<p>All equipment prices are as per GS-00C-01911.</p> <p>If LNK Corp desires to purchase utilizing the Federal Supply Schedule, a letter from the Contracting Officer (Gov't) in authorization must be provided.</p> <p>The hardware configurations shown reflect optimal (not lowest cost) configurations. An 8635-NA (without interleaving - UNAMACE systems) can be substituted for the 8635-NB at lower cost.</p>								

\*Model/Feature Defined in Data General Price List

B-4

See Page 1 for Signature

07-898011464

CUSTOMER COPY

# PERKIN-ELMER

TO • LMK CORPORATION  
 • 4321 Hartwick Road  
 • College Park, Maryland 20740  
 •

## QUOTATION

FROM • PERKIN-ELMER  
 • COMPUTER SYSTEMS DIVISION  
 • 1764 Old Meadow Lane  
 • McLean, Virginia 22101  
 •

QUOTATION NUMBER:  
 2070-09-428A

PLEASE REFER TO THIS QUOTATION NUMBER ON ALL CORRESPONDENCE AND ORDERS

DATE 1/26/81  
 VALID UNTIL 4/26/81

ITEM	PRODUCT NUMBER	DESCRIPTION	UNIT LIST PRICE	QTY.	TOTAL LIST PRICE	DISC.	TOTAL NET PRICE	MAINTENANCE
1.	M32-260	Model 3220 Package System includes: .3220 Processor with 524,288 bytes MOS Memory .8 sets of 16, 32 bit general registers .Power Fail/Auto Restart .Memory Access Controller .OS/32 Bootloader .Loader Storage Unit .System Console .Quantity three Video - Model 550 .3200 Selector Channel .2 Line Comm Mux .8 Line Comm Mux .I/O Expansion Chassis .56" Beige Cabinet .MSM 80F Disc System with controller .67MB Formatted Fixed Media .9 Track, 800/1600 CPI 75 IPS Tape System with Controller and Cabinet .Memory Access Controller .Battery Backup	75,100.	1	75,100.	21	59,329.	783.
2.	M46-691	MSM 80F Disc System with Controller 67MB Formatted Fixed Media	15,200.	1	15,200.	21	12,008.	150.

BY *Andy Chalkley*

AUTHORIZED REPRESENTATIVE  
 THE PERKIN-ELMER CORPORATION  
 COMPUTER OPERATIONS

# PERKIN-ELMER

## QUOTATION

TO . . . LMK Corporation  
 . . .  
 . . .  
 . . .

FROM . . . Perkin-Elmer  
 . . .  
 . . .  
 . . .

QUOTATION NUMBER:  
 2070-09-428A

PLEASE REFER TO THIS QUOTATION NUMBER ON ALL CORRESPONDENCE AND ORDERS

DATE \_\_\_\_\_  
 VALID UNTIL \_\_\_\_\_

ITEM	PRODUCT NUMBER	DESCRIPTION	UNIT LIST PRICE	QTY.	TOTAL LIST PRICE	DISC.	TOTAL NET PRICE	MAINTENANCE
3.	M46-495	Expansion Tape, 9 Track, 800/1200 CFI 75IPS with Cabinet	9,625.	1	9,625.	21	7,603.	100.
4.	M32-004	Floating Point Hardware	5,600.	1	5,600.	21	4,424.	40.
5.	M32-001	1,024 Bytes Cache Memory	3,500.	1	3,500.	21	2,765.	5.
6.	M32-010	3200 Selector Channel	1,750.	1	1,750.	21	1,382.	15.
7.	M46-221	CF 120 Matrix Printer	3,500.	1	3,500.	21	2,765.	50.
8.	M46-233	CP 120 Printer Interface	900.	1	900.	21	711.	20.
		TOTAL					90,998.	1,053.

BY *[Signature]*

AUTHORIZED REPRESENTATIVE  
 THE PERKIN-ELMER CORPORATION  
 COMPUTER OPERATIONS

# PERKIN-ELMER

TO • LNK Corporation  
 • 4321 Hartwick  
 • College Park, Maryland  
 •  
 • Attn: Russ Smith

## QUOTATION

FROM • Perkin-Elmer Corporation  
 • 1764 Old Meadow Lane  
 • Polk Bldg., Suite 1  
 • McLean, Virginia 22102  
 •

QUOTATION NUMBER:  
 2070-09-427

PLEASE REFER TO THIS QUOTATION  
 NUMBER ON ALL CORRESPONDENCE  
 AND ORDERS

DATE: 7/24/80

VALID UNTIL: 10/24/80

ITEM	PRODUCT NUMBER	DESCRIPTION	UNIT LIST PRICE	QTY.	TOTAL LIST PRICE	DISC.	TOTAL NET PRICE	MAINTENANCE
1	M32-401	Hardware: Model 3242 Processor with 524,288 bytes of MOS ECC memory. 180-264 volts, 47-63 Hz includes: • Expansion to 16MB of real memory • 8 Sets of 16 32 bit general registers • Power fail auto restart • Loader storage unit • Universal Clock • OS/32 Bootloader • 8Kb 4-way set associative CACHE • Model 550 CRT • 2 Line Comm. Mux. • 3240 Power Supply • Two 56" Biege Cabinets		1	\$111,000	21%	\$ 87,690	720
2	M32-423	Floating Point Processor		1	\$ 9,500	21%	\$ 7,505	60
3	M32-425	2K Words of 3240 Writable Control Store W/development software on 1600 CPI Tape.		1	\$ 7,500	21%	\$ 5,925	40
4	M46-691	MSM 80F Disc System with Controller, Formatted 67 MB Fixed Media.	\$ 15,200	2	\$ 30,400	21%	\$ 24,016	150

BY: *Ruby Anderson*  
 AUTHORIZED REPRESENTATIVE  
 THE PERKIN-ELMER CORPORATION  
 COMPUTER OPERATIONS

# PERKIN-ELMER

TO • LNK Corporation  
 • 4321 Hartwick  
 • College Park, Maryland  
 •  
 •

## QUOTATION

FROM • Perkin-Elmer Corporation  
 • 1764 Old Meadow Lane  
 • Polk Bldg., Suite 1  
 • McLean, Virginia 22102  
 •

QUOTATION NUMBER:  
 2070-09-427

PLEASE REFER TO THIS QUOTATION NUMBER ON ALL CORRESPONDENCE AND ORDERS

DATE: 7/24/80  
 VALID UNTIL: 10/24/80

ITEM	PRODUCT NUMBER	DESCRIPTION	UNIT LIST PRICE	QTY.	TOTAL LIST PRICE	DISC.	TOTAL NET PRICE	MAINTENANCE
5	M32-010	3200 Selector Channel	\$ 1,750	2	\$ 3,500	21%	\$ 2,765	15
6	M46-494	9 Track, 800/1600 CPI, 75 IPS Mag Tape System with Controller and Cabinet.	\$19,100	1	19,100	21%	\$15,089	140
7	M46-495	9 Track, 800/1600, 75 IPS Expansion Drive, w/Cabinet	\$ 9,625	1	\$ 9,625	21%	\$ 7,603	100
8	M46-221	CP 120 Matrix Printer	\$ 3,500	1	\$ 3,500	21%	\$ 2,765	50
9	M46-233	CP Printer Interface	\$ 900	1	\$ 900	21%		10
		Totals:			\$195,025		\$154,069	1,285
10	S80-016-ABC	SOFTWARE: OS/32			\$ 6,000		\$ 6,000	
122	S80-016-ABC	FORTAN VII			\$ 6,000		6,000	
13	S90-028-071	FORTAN Enhancement Package			600		600	
		Totals:			\$ 12,600		\$ 12,600	

BY: *Paul Anderson*  
 AUTHORIZED REPRESENTATIVE  
 THE PERKIN-ELMER CORPORATION  
 COMPUTER OPERATIONS

# PERKIN-ELMER

TO • LNK Corporation  
 • 4321 Hartwick  
 • College Park, Maryland  
 •  
 • Attn: Russ Smith

## QUOTATION

FROM • Perkin-Elmer Corporation  
 • 1764 Old Meadow Lane  
 • Polk Bldg., Suite 1  
 • McLean, Virginia 22102  
 •

QUOTATION NUMBER:  
 2070-09-424

PLEASE REFER TO THIS QUOTATION NUMBER ON ALL CORRESPONDENCE AND ORDERS

DATE: July 21, 1980  
 VALID UNTIL: Oct. 21, 1980

ITEM	PRODUCT NUMBER	DESCRIPTION	UNIT LIST PRICE	QTY.	TOTAL LIST PRICE	DISC.	TOTAL NET PRICE	MAINTENANCE
1	M32-401	Hardware: Model 3242 Processor with 524,288 bytes of MOS EOC memory, 180-264 volts, 47-63 Hz includes: • Expansion to 16MB of real memory • 8 Sets of 16 32 bit general registers • Power fail auto restart • Loader storage unit • Universal Clock • OS/32 Bootloader • 8Kb 4-way set associative CACHE • Model 550 CRT • 2 Line Comm. Mux. • 3240 Power Supply • Two 56" Biege Cabinets		1	\$111,000	21%	\$ 87,690	720
2	M32-430	1,048,576 Bytes MOS/EOC Memory		1	\$ 23,900	21%	\$ 18,881	240
3	M32-423	Floating Point Processor		1	\$ 9,500	21%	\$ 7,505	60
4	M32-425	2K Words of 3240 Writable Control Store w/development software on 1600 CPI TAPE		1	\$ 7,500	21%	\$ 5,925	40
5	M46-691	MSM 80F Disc System with Controller, Formatted 67 MB Fixed Media.	\$ 15,200	2	\$30,400	21%	\$ 24,016	150

Page 1 of 2

BY



AUTHORIZED REPRESENTATIVE  
 THE PERKIN-ELMER CORPORATION  
 COMPUTER OPERATIONS

# PERKIN-ELMER

TO • INK Corporation  
 • 4321 Hartwick  
 • College Park, Maryland  
 •  
 •

## QUOTATION

FROM • Perkin-Elmer Corporation  
 • 1764 Old Meadow Lane  
 • Polk Bldg., Suite 1  
 • McLean, Virginia 22102  
 •

QUOTATION NUMBER:  
 2070-09-424

PLEASE REFER TO THIS QUOTATION NUMBER ON ALL CORRESPONDENCE AND ORDERS

DATE: July 21, 1980  
 VALID UNTIL: Oct. 21, 1980

ITEM	PRODUCT NUMBER	DESCRIPTION	UNIT LIST PRICE	QTY.	TOTAL LIST PRICE	DISC.	TOTAL NET PRICE	MAINTENANCE
6	M32-010	3200 Selector Channel	\$1,750	2	\$3,500	21%	\$2,765	15
7	M46-494	9 Track, 800/1600 CPI, 75 IPS Mag Tape System with Controller and Cabinet	\$19,100	1	\$19,100	21%	\$15,089	140
8	M46-495	9 Track, 800/1600, 75 IPS Expansion Drive, w/Cabinet	\$9,625	1	\$9,625	21%	\$7,603	100
9	M46-221	CP 120 Matrix Printer	\$3,500	1	\$3,500	21%	\$2,765	50
10	M46-233	CP Printer Interface	900	1	900	21%		10
		Totals			\$218,925		\$172,950	1,525
		SOFTWARE:						
11	S80-016-ABC	OS/32			\$6,000		\$6,000	-
12	S80-016-ABC	FORTRAN VII			\$6,000		\$6,000	-
13	S90-028-071	FORTRAN Enhancement Package			\$600		\$600	
		Totals			\$12,600		\$12,600	

BY *Randy Anderson*  
 AUTHORIZED REPRESENTATIVE  
 THE PERKIN-ELMER CORPORATION  
 COMPUTER OPERATIONS



# DIGITAL EQUIPMENT CORPORATION

PHONE AC 617 897-5111 TWX 710-347-0212-CABLE DIGITAL MAYN. TELEX 94-84-57

## QUOTATION

QUOTATION NUMBER
C-104-104-8062
PLEASE REFER TO THIS QUOTATION NO IN ALL CORRESPONDENCE AND ORDERS

DATE 22 September, 1980  
REFERENCE GS-00C-01892

DISCOUNT AGREEMENT NO \_\_\_\_\_  
NEAREST DIGITAL SALES OFFICE

PRODUCT LINE CODE

TO L nk Corporation  
4321 Hartwick Rd  
Suite 321  
College Park, MD 20740  
Attn: Russ Smith

8301 Professional Place  
Landover, Maryland 20785

THANK YOU FOR YOUR INQUIRY. WE ARE PLEASED TO QUOTE AS FOLLOWS

ITEM	QTY	MODEL NO	DESCRIPTION	UNIT PRICE	NET AMOUNT
1	1	11X44-CA	256KB ECC MOS, H9642, TUJ8, 120V	\$ 27,700	
2	1	LA38-HA	DECwriter II	\$ 1,700	
3	1	FP11-F	FPP for 1144	\$ 3,100	
4	1	RJM02-AA	67MB Disk & Controller	\$ 25,700	
5	1	RM02-AA	67MB Disk Drive	\$ 19,300	
6	1	TJE16-EA	800/1600 bpi 45 ips tape dr.	\$ 20,200	
7	1	TE16-AE	800/1600 bpi,45ips tape drive	\$ 12,800	
8	1	H9642-DB	Expansion Cabinet without end panels	\$ 1,300	
9	1	BA11-KW	Expansion Box	\$ 3,200	
10	1	EC03M-25	Null Modem Cable 1	\$ 60	
11	1	DL11-WE	EIA Async Line RT Clock	\$ 820	
12	1	QJ737-AD	RSX11-M	\$ 7,800	
13	1	QP230-AD	Fortran IV/RSX 11M	\$ 1,000	
14	1	DD11-DK	DD11-D, 2-SU for BA11-K	\$ 860	
Subtotal:				\$125,540	
Less GSA Discount				\$ 18,831	
				\$106,709	

This quotation shall remain firm for 60 days from the date hereof, unless modified in writing by Digital Equipment Corporation prior to our acceptance of your contract offer. This quotation is subject to credit approval and is governed by the Digital Equipment Corporation standard terms and conditions of sale appearing on the reverse hereof and of the terms as noted above and attached hereto.

- 1 \* Software consulting or resident terms and conditions
- 2 \* Computer Special Systems terms and conditions
- 3 \* DEC system purchase and sale agreement
- 4 \* Digital Component Grid terms and conditions of sale
- 5 \* Discount agreement between purchaser and Digital as filed in above

Please for all requested information on the ORDER FORM, page 2 of this form, and forward to your Digital Sales Office, as shown on contract offer. Any contract resulting from this quotation must be accepted at Digital's corporate offices by a duly authorized representative of Digital Equipment Corporation. Insurance will be provided on property while in transit and a charge of \$500 per \$100,000 of equipment value will be made unless instructions to the contrary are clearly stated on the face of the Purchaser's order.

SUB TOTAL	\$106,709.00
PLUS INSTALLATION	
PLUS INSURANCE	
NET TOTAL AMOUNT	

ESTIMATED DELIVERY SCHEDULE

*Dale Vogel*  
Dale Vogel

QUOTATION PREPARED BY

PAGE 1 OF 1

QUOTATION



DIGITAL EQUIPMENT CORPORATION

PHONE AC 617 897-5111 TWX 710-347-0212-CABLE DIGITAL MAYN. TELEX 94-84-57

QUOTATION NUMBER

C-104-104-8061

PLEASE REFER TO THIS QUOTATION NO IN ALL CORRESPONDENCE AND ORDERS

DATE 12 September 1980

REFERENCE \_\_\_\_\_

DISCOUNT AGREEMENT NO \_\_\_\_\_

NEAREST DIGITAL SALES OFFICE \_\_\_\_\_

PRODUCT LINE CODE \_\_\_\_\_

TO LNK Corporation  
4321 Hartwick Rd.  
Suite 321  
College Park, MD 20740  
Attn: Russ Smith

8301 Professional Place  
Landover, Maryland 20785

THANK YOU FOR YOUR INQUIRY. WE ARE PLEASSED TO QUOTE AS FOLLOWS

ITEM	QTY	MODEL NO	DESCRIPTION	TERM	MONTHLY	430 OR	DISC	UNIT PRICE	NET AMOUNT
				9	MAINT	400 OR	4		
						400 OR			
						400 OR			
1	1	11/34A-YC	11/34A CPU, 256KB MOS Memory; Serial Line Interface					\$15,300	\$15,300
2	1	FF11-A	Floating Point Processor					\$ 2,900	\$ 2,900
3	2	RJM02-AA	67MB Disk & Controller					\$24,000	\$48,000
4	1	LA35-HA	DECwriter II - RS232					\$ 1,700	\$ 1,700
5	1	TE16-EA	800/1600 bpi 45 ips tape drive & controller.					\$20,200	\$20,200
6	1	TE16-AE	800/1600 BFI tape Drive					\$12,800	\$12,800
7	1	H960-CA	Cabinet					\$ 1,575	\$ 1,575
8	1	EC03M-25	Null Modem Cable					\$ 60	\$ 60
9	2	BA11-KE	Expansion Box					\$ 3,000	\$ 6,000
10	1	RJ737-AE	RSX11M					\$ 7,800	\$ 7,800
11	1	PE30-AL	FORTAN IV/RSX11M					\$ 1,000	\$ 1,000
TOTAL:									\$117,335.00
(GSA) 15% Discount									<u>\$ 17,600.25</u>
									<u>\$ 99,734.75</u>
SUB TOTAL									
PLUS INSTALLATION									
PLUS INSURANCE									
NET TOTAL AMOUNT									

This quotation shall remain firm for 60 days from the date hereof unless modified in writing by Digital Equipment Corporation prior to the acceptance of your contract offer. This quotation is subject to credit approval and is governed by the Digital Equipment Corporation standard terms and conditions of sale appearing on the reverse hereof and/or the terms as noted above and attached hereto.

- 1 \* On-site consulting
- 2 \* Computer Special Systems terms and conditions
- 3 \* DEC system purchase and sale agreement
- 4 \* Digital Components Group terms and conditions of sale
- 5 \* Discount agreement between purchaser and Digital as filed above

Please be sure to request a copy of the ORDER FORM, page 2 of this form, and forward to your Digital Sales Office as your contract offer. All quotations from this quotation must be accepted at Digital's corporate offices by a duly authorized representative of Digital Equipment Corporation. Insurance will be provided on property while in transit and a charge of \$50 per \$100.00 of equipment value will be made unless instructions to the contrary are clearly stated on the face of the Purchaser's order.

ESTIMATED DELIVERY SCHEDULE SUBJECT TO MODIFICATION BY DIGITAL: \_\_\_\_\_

QUOTATION PREPARED BY Dale Vogel *Dale Vogel*

PAGE 1 OF 1