(8)

AD-A200 325

RD&E CENTER

Technical Report

TACOM-TR-

No. 13287

CREATING AND ANIMATING VEHICLE GRAPHICS

JULY 1988

Christopher E. Morlan
U.S. Army Tank-Automotive Command
ATTN: AMSTA-RYA
By Warren, MI 48397-5000

U.S. ARMY TANK-AUTOMOTIVE COMMAND
RESEARCH, DEVELOPMENT & ENGINEERING CENTER
Warren, Michigan 48397-5000

NOTICES

This report is not to be construed as an official Department of the Army position.

Mention of any trade names or manufacturers in this report shall not be construed as an official endorsement or approval of such products or companies by the U.S. Government.

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

UNCLASSIFIED
URITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188
Exp Date Jun 30, 1986

1a. REPORT SECURITY CLASSIFICATION nclassified	1b RESTRICTIVE MARKINGS None
2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release. Distribution is unlimited.
2b DECLASSIFICATION/DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S) 13287
6a. NAME OF PERFORMING ORGANIZATION US Army Tank-Automotive Command 6b. OFFICE SYMBOL (If applicable) AMSTA-RYA	7a. NAME OF MONITORING ORGANIZATION U.S. Army Tank-Automotive Command
6c. ADDRESS (City, State, and ZIP Code) Warren, MI 48397-5000	7b. ADDRESS (City, State, and ZIP Code) Warren, MI 48397-5000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION US Army Tank-Automotive Command 8b. OFFICE SYMBOL (If applicable) AMSTA-RYA	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code) Warren, MI 48397-5000	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO

11. TITLE (Include Security Classification)

Creating and Animating Vehicle Graphics (U)

12. PERSONAL AUTHOR(S)
Christopher E. Morlan

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM May 87 TO Aug 87	JUL 88	16

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	MOVIE.BYU, Lexidata System, VAS Controller, PLOT Program

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This report is intended to describe a standardized method of creating graphics files for use in vehicle animations. It also details the use of the programs required to do actual animations, including the program PLOT, which provides the use of graphs in conjunction with animations.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION Unclassified
22a NAME OF RESPONSIBLE INDIVIDUAL David S. Rohweder	22b TELEPHONE (Include Area Code) (313)574-6365 22c OFFICE SYMBOL AMSTA-RYA

D FORM 1473, 84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

1.0. INTRODUCTION

This report, prepared by the Analytical and Physical Simulation Branch of the Tank-Automotive Technology Directorate, of the U.S. Army Tank-Automotive Command, describes the conventions and methods used to create and animate vehicle graphics. It is intended to be used in conjunction with the programs MOVIE.BYU, ANIMATE, and PLOT, and the VAS controller, the VAX 8800 computer, and the Lexidata SOLIDVIEW device. It is divided into the following sections:

- Conventions,
- Creating vehicle graphics,
- Animation methods, and
- Use of PLOT program.

2.0. OBJECTIVE

It is the goal of this report to make the user familiar with and knowledgeable of the methods available for creating and animating vehicle graphics. To this end, the use of MOVIE.BYU, ANIMATE, and PLOT are described in detail. It is recommended, however, that before attempting to use the available software the user read at least the relevant sections of the MOVIE.BYU manual and review the documentation of the vehicle graphics already created.

3.0. CONCLUSIONS

Using this report and existing documentation of vehicle graphics, the user can create and animate either existing vehicle graphics or files of their own design.

4.0. RECOMMENDATIONS

It would greatly reduce the time required to test an animation if a method were employed whereby files including animation and graphing data could be used.

5.0. DISCUSSION

5.1. Conventions

5.1.1. Origin. Using vehicle drawings, an origin must be selected. It should be even with the rear, the bottom, and the center of the vehicle. In other words, when looking at the left-side view of the vehicle, the

origin should be in the lower right-hand corner of the drawing, halfway into the page.

5.1.2. Axes. For consistency, the axes should always be oriented in the same manner for all vehicles. A right-hand coordinate system is generally used. The positive X axis should extend out the right side of the vehicle, the positive Y out the front, and the positive Z toward the top. Looking at a left-side view of the vehicle, the X axis is into the page, the Y axis is to the left, and the Z axis extends upward on the drawing.

5.1.3. Coordinates. The coordinates of all points are, by convention, measured in inches.

5.2. Creating Vehicle Graphics

5.2.1. Noncurved Parts. Vehicle parts are created using a MOVIE.BYU program called UTILITY. When running this program, noncurved parts are generated by entering GEOMETRY CHANGE (since only a four-letter abbreviation is required for all commands, GEOM CHAN will suffice). In this mode the next command tells UTILITY what the user wishes to change. The command COORDINATE (COOR) invokes a mode in which the coordinates of the geometry file may be altered (in this case, entered). The program then prompts for the number of nodes (points) to be created. For example, a cube would be made up of eight nodes. More complex shapes would require more nodes. The X,Y,Z coordinates of each node (in feet) are then entered, preceded by the node number. When this process is complete, the user should, while still in the GEOM CHAN mode, enter ELEMENT (ELEM). UTILITY is now ready to accept element definitions in terms of nodes. The next prompt requests the number of nodes per element. A cube would have four nodes per element; the more complex the part, the mode nodes per element.

5.2.2. Adding Elements. Answering the next prompt with ADD (A) tells UTILITY you wish to add elements. When defining elements, the group number (1) is entered, followed by the nodes to be connected to form the element. Nodes should be entered according to the right-hand rule. Accordingly, the nodes are entered in a counterclockwise order when viewing the element from outside. This corresponds to having the thumb of the right hand facing outward and listing the nodes in the order indicated by the direction in which the fingers are curling (counterclockwise).

5.2.3. Curved Parts. To create circular or cylindrical parts, another method must be used. This method involves another mode of UTILITY, MAKE VOLUME REVOLVE (MAKE VOLU REVO). This method consists of defining a two dimensional shape in the X-Y plane and then revolving it about the X axis. The first prompt to be answered (hit RETURN for extraneous prompts) requests the number of nodes on the shape to be created and the angle it is to be revolved. Four nodes revolved 360 degrees would form a solid cylinder, such as an axle. More nodes are required for more

complex parts. Three hundred sixty degrees is the standard revolution for solid parts. Next, the X,Y,Z coordinates (feet) are entered. The next prompt is for the number of circumferential elements (elements around the circumference of the circle) to be created. This parameter determines the smoothness of the curve to be formed (12-16 elements will form a fairly smooth curve). Using too many elements will unnecessarily slow the drawing of the part, while using too few will cause the curve to look jagged. A compromise must be found, depending on the size, shape, and importance of the part.

5.2.4. Saving Geometry Files. After creating a part using either of the above methods, the geometry file must be saved. This is done by changing to the GEOMETRY WRITE (GEOM WRIT) mode. The program will then prompt for the geometry filename. After choosing a name having the extension .GEO, the program may be terminated by typing EXIT.

5.2.5. Cube Example. The following is a step-by-step example of the creation of a simple part using UTILITY in the GEOM CHAN mode. The part to be created is a basic cube. The first step is to run the program and then enter GEOM CHAN COOR. The number of nodes to be entered is 8, as follows: 1 0 0 0, 2 0 0 2, 3 2 0 2, 4 2 0 0, 5 0 2 0, 6 0 2 2 , 7 2 2 2 , 8 2 2 0. This sequence defines the 8 nodes of a cube which is 2 feet on a side. The elements must be defined next, each consisting of 4 nodes. They are entered as follows: 1 4 3 2 1, 1 5 6 7 8, 1 1 2 6 5, 1 2 3 7 6, 1 3 4 8 7, 1 4 1 5 8. This defines the 6 faces of the cube, all belonging to group 1, each consisting of a different combination of four corner nodes. GEOM WRIT is entered next, the file being named CUBE.GEO. Pictures of this part, both in draw mode showing the nodes and in view mode showing it as a solid, are shown in Figures 5-1 and 5-2.

5.2.6. Cylinder Example. The following is an example describing a cylindrical shape. After running UTILITY, MAKE VOLU REVO is entered. Four nodes are to be used and the shape is to be revolved 360 degrees. The four nodes are then entered as follows: 0 0 0, 0 0 6, 3 0 6 , 3 0 0. The only other prompt to be answered is the one requesting the number of circumferential elements, 16. This part must also be saved by typing GEOM WRIT, the file being named CYL.GEO. Pictures of this part in both modes are in Figures 5-3 and 5-4.

5.2.7. Rotating Parts. Once the geometry file of a part is completed, it is often necessary (especially for parts made using MAKE VOLU REVO) to orient it properly with respect to the axes. This is done by using a program called ROTATE. This program prompts first for the input and output filenames, which must be supplied. The next step is to input the angles of rotation about the X,Y,Z axes. ROTATE then creates an output file containing the rotated data and ends.

5.2.8. Translating Parts. Once they are created and oriented correctly, parts must often be translated (moved) to their proper positions relative to the rest of the vehicle. This process is performed

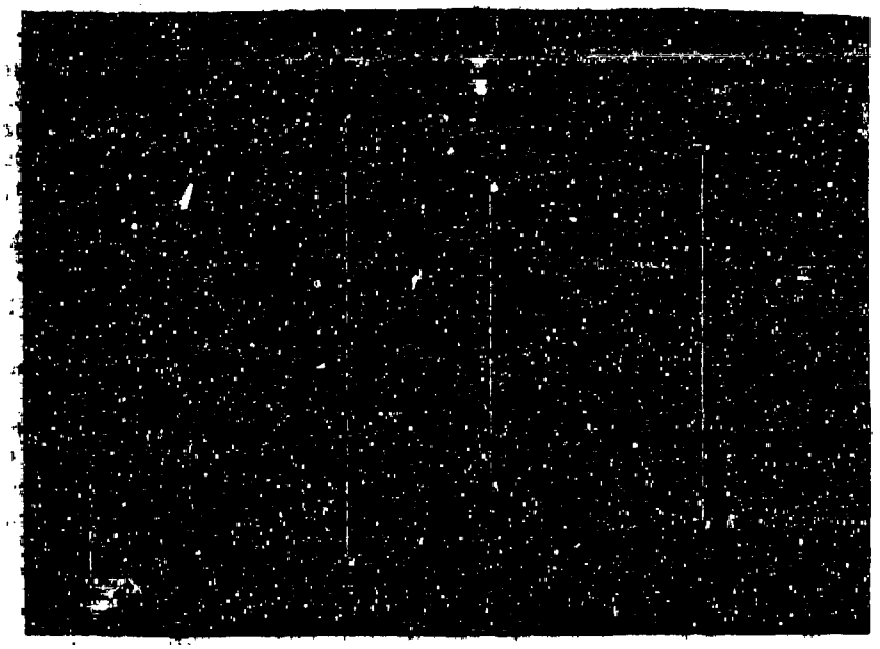

Figure 5-1. Draw Mode Picture of Cube (Observe numbered nodes)

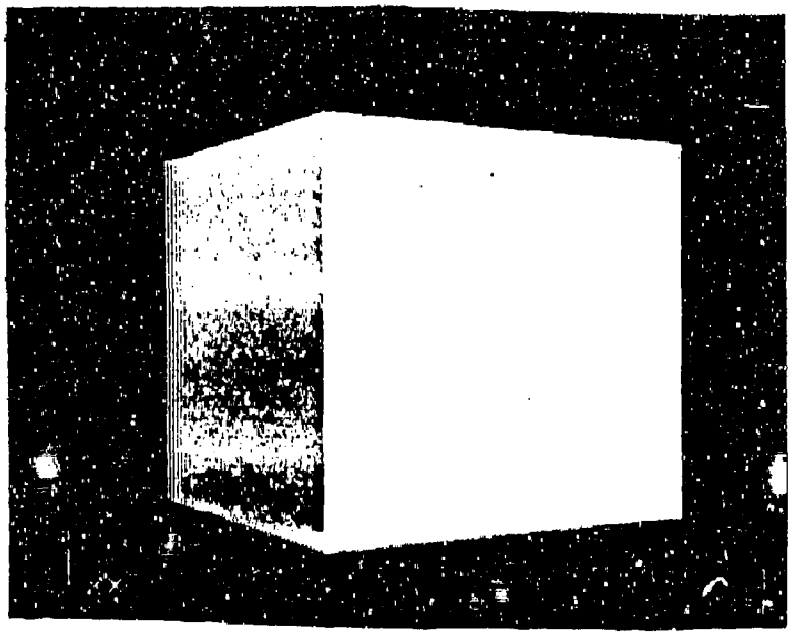

Figure 5-2. View Mode Picture of Cube

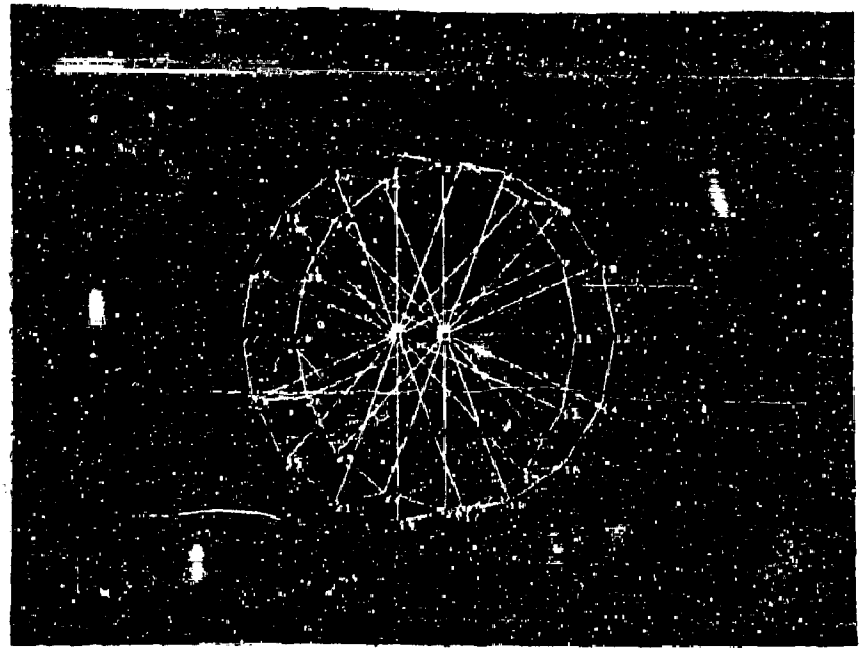

Figure 5-3. Draw Mode Picture of Cylinder

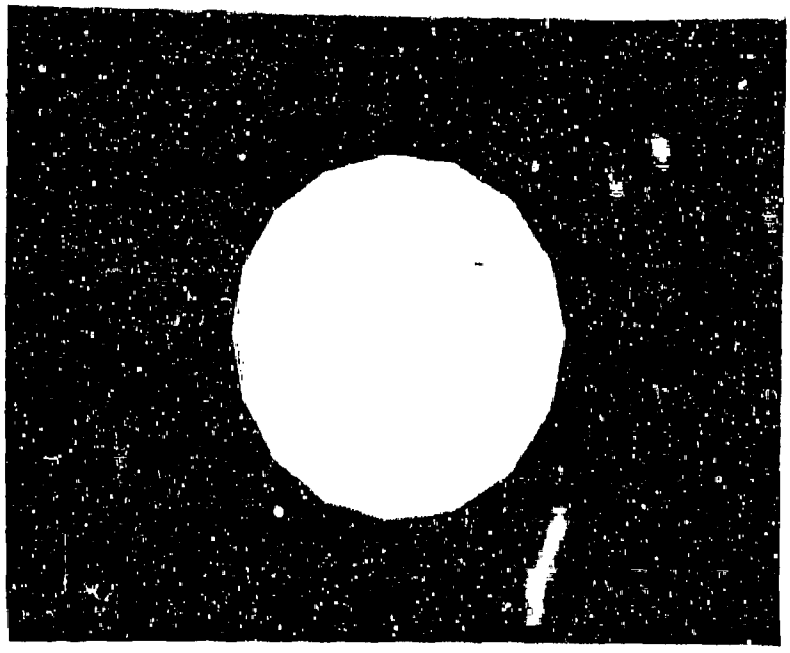

Figure 5-4. View Mode Picture of Cylinder

using a program called TRANSLATE. This program also prompts for the input and output geometry filenames. After these are entered, the program prompts for the translation amounts (feet) along the axes. For example, inputing 2 1 2 would move the entire part 2 feet toward the right of the vehicle (along the X axis), 1 foot toward the front of the vehicle (along the Y axis), and 2 feet up (along the Z axis). TRANSLATE creates an output file containing the translated data and ends.

5.2.8.1. Merging Parts. Once all of the parts required to form a vehicle are created and placed in the correct orientations and locations, they must be combined. In order to maintain a reasonable number of parts, it is often desirable to combine several similar parts to form one larger part. This is accomplished using the program MERGER to combine the geometry part files into one larger geometry part file. The program prompts for a command, the proper response being MERGE (M). It then prompts for all the input filenames (% halts this process) and finally, the output filename. An output filename without an extension should be used, since MERGER creates three new files with extensions .GEO, .INF, .BOD.

5.2.8.2. The MERGER Program. The MERGER program will create a new geometry file containing the coordinates and elements of the merged parts. It will, however, still consider the components to be separate parts. To modify this, UTILITY must be used once again. The GEOMETRY READ (GEOM READ) mode must be used and the filename entered. The program will then prompt <CHANGES?>. A response of YES (Y) will return the prompt <NUMBER OF GROUPS>. To combine several parts into one, enter 1. The next question is <NUMBER OF ELEMENTS>. Simply enter the total number of elements in all parts being combined. Then save the file under a new name using GEOM WRIT. Once the total number of parts has been reduced to a reasonable figure (usually 20 or less), the actual vehicle may be assembled. This is done by using MERGER again and inputing the filenames for all the parts to create one large vehicle file.

5.3. Animation Methods

5.3.1. Viewing Vehicle. There are two animation programs available, ANIMATE2 and ANIMATE3. For the most part they are identical. With either program, a series of data must be entered. The first three prompts are for geometry, body and DADS filenames. The geometry and body files are created by the MERGER program, but the DADS file must be created separately. DADS is an independent program, used to calculate the location of the various bodies on the vehicle at different times during the animation. Colors are then defined, using the command COLOR, by three components (0-1) for Red, Blue and Green. Parts are then colored by assigning color numbers to part numbers. The vehicle may be positioned on the screen using DISTANCE (DIST) and ROTATE (ROTA) commands which are fairly self-explanatory. To display the vehicle, the command VIEW is then entered. All the above parameters may be entered into a .SET file for ease of reproduction. Details on the use of .SET files may be found in the MOVIE.BYU documentation.

5.3.2. ANIMATE3. ANIMATE3 is a program which allows the user to operate the VAS (Video Animation Sysem) from a terminal, thereby providing complete control over the animation process from one location. Once the vehicle is properly displayed, the animation process is begun by typing the command ANIM, which produces a series of prompts. The user must then enter the number of frames per record for the first frame of the animation. When played back, 30 frames of animation cover 1 second of actual time. Therefore, 60 frames per record for the first frame would provide a 2-second leader.

5.3.3. Scene Number. The next prompts ask for scene number (arbitrary, begin with 1) and whether or not it is new (yes=1). Next, the program asks for the first frame number of the scene (1). Next, the user is asked for the starting DADS frame number. This parameter allows the user to begin the animation at any point in the DADS file. The following prompt allows for skipping DADS frames, which can further accelerate the animation process. If this option is selected, any number of DADS frames may be skipped between each one used. The next prompt inquires as to whether or not the user wants graphing during the animation. This topic will be dealt with in a later section of this report.

5.3.4. DADS Steps. This program next asks how many DADS time steps are to be used. This is merely the number of data points the user wishes to include in the animation. If the total number is unknown, merely enter 9999 and the program will stop when it reaches the end of the DADS file. Lastly, the number of frames per record for the rest of the animation is requested. If the DADS data is calculated every 1/30th of a second and the animation is to be done in real time, simply enter 1. However, if the data is less frequent than that or if the animation is to be done in slow motion, then a higher number should be entered. The animation should then run to completion without further input from the user.

5.3.5. ANIMATE2. ANIMATE2 is a less complex version of ANIMATE3. It does not allow for any manipulation of DADS data, but merely allows programming of the VAS directly (not from a terminal). Again, the vehicle must first be properly positioned and displayed before the animation process may begin. Once this is done, the Program button on the VAS controller must be depressed. The first four prompts that were answered for ANIMATE3 must be answered from the VAS keypad. Once this is done, the command ANIM is entered on the terminal to begin the animation. After the first frame is recorded, the number of frames per record may be altered using the Frame Change button on the VAS. Once this is done, the animation should run to completion on its own.

5.4. Using PLOT

If, when using ANIMATE3, the user indicates that graphing is desired, then the program PLOT is engaged. This will initiate another series of prompts. The first one requests the number of graphs desired (1 or 2). The data filename is then required. Data files may contain as many

header lines as necessary, provided they do not begin with numbers. The data itself may be in any format, but must contain two numbers (X and Y) on each line. All files should have the extension .DAT. Just the filename itself should be entered. Next, the labels for the X and Y axes must be entered. The labels must contain no more than eight characters each. They may be lower case or capitals. If lower case characters are entered, however, the labels themselves will be capitalized. The origin coordinates must be entered next. The screen coordinates are 640 (X) by 512 (Y) with the origin being in the upper left corner. Accordingly, an origin of 100, 100 usually works well. The user is then asked to input the axis lengths, with 80 creating a good-sized graph. The data will be scaled to fit the graph, regardless of axis lengths. Once all the above prompts are answered, the animation process will begin. A graph will appear in every frame, including the most recent data point.

DISTRIBUTION LIST

	Copies
Commander Defense Technical Information Center Bldg. 5, Cameron Station ATTN: DDAC Alexandria, VA 22304-9990	12
Manager Defense Logistics Studies Information Exchange ATTN: AMXMC-D Fort Lee, VA 23801-6044	2
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-DDL Warren, MI 48397-5000	2
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-CF (Mr. G. Orlicki) Warren, MI 48397-5000	1
Director U.S. Army Materiel Systems Analysis Activity ATTN: AMXSY-MP (Mr. Cohen) Aberdeen Proving Ground, MD 21005-5071	1
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-RYA Warren, MI 48397-5000	12