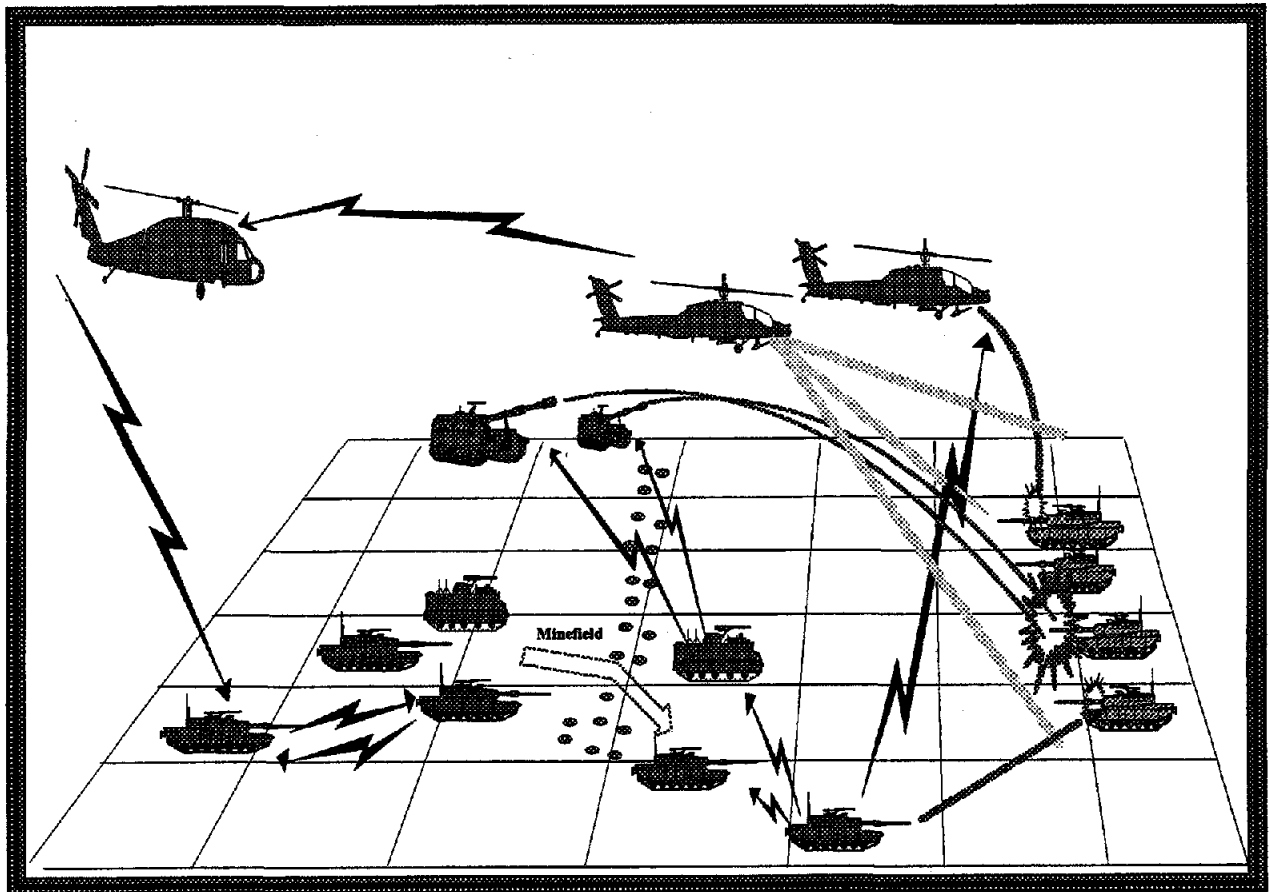


The Tank Battle of the 21st Century at TEXCOM

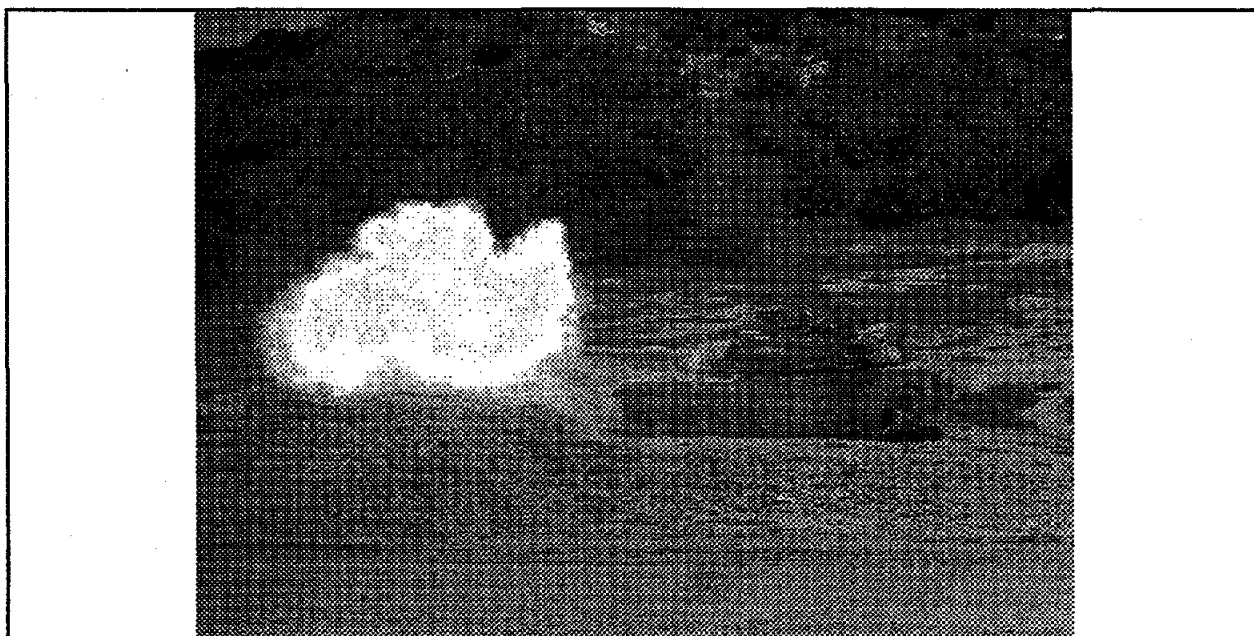
What does SAS have to do with tank battles? In the 21st century? In the next 45 minutes, I'll connect these concepts. We will talk about military weapon systems for the future, the mission of US Army testing, the battlefield of the 21st century, and how SAS/GRAPH/ANNOTATE was used to display a tank battle over time.

The battlefield of the 21st century will be a digital environment , in which the tank will receive information from satellite and ground based sensors which will be extensively shared. This will enable tanks to engage at greater ranges, within the limits of the tank's weapon systems. We will see more extensive sharing of targets and information through technology.



*The Test and Experimentation Command (TEXCOM) at Fort Hood, Texas plays an important role in moving the US Army onto the battlefield of the 21st century. Please watch the video of the TEXCOM mission. ***9 minutes of the TEXCOM command briefing***. With the high costs of weapon systems, operational testing is essential. In an operational test, the weapon system is given to representative soldiers to use in a combat environment. The resulting effects are then analyzed. Something can look wonderful on the drawing board but show serious flaws and oversights when it is given to soldiers to use in a simulated combat environment. TEXCOM takes pride in its role as consumer advocate for the US soldier. As an example, TEXCOM conducted extensive testing of the Abrams M1A2 tank with its sophisticated electronics designed to fight on battlefield of the 21st century.*

*The US Army Abrams M1A2 main battle tank in combat is the subject of the SAS graphic. Please watch this video of the Abrams tank in live fire and in instrumented testing. The laser instrumentation promotes realism and provides data for analysis. This will be described in more detail later. ***3 minutes of Abrams music video***. This vehicle can move at 45 miles per hour and fire its main gun on the move. At about \$7 million per tank, the Abrams thermal imaging allowed it to see at night and through the smoke of the Kuwait oil fires. This tank can digitally share battlefield targets.*



*Let's talk about "The Tank Battle of the 21st Century at TEXCOM". *** slides & hard copy mounted *** This graphic was produced by SAS/GRAPH/ANNOTATE. It shows a tank battle over time. The time scale is at the bottom of the graph. TEXCOM instrumented battles rarely last more than 30 minutes and more often are 5 minutes of intense combat. The horizontal GREEN line divides the BLUE (friendly) forces and the RED (enemy) forces. The horizontal BLACK line shows "killed" vehicles. The RED vertical lines are enemy firings. The BLUE vertical lines are friendly firings. These graphics even show fratricide (FRIENDLY killing FRIENDLY)! One graph clearly shows what happened in the tank battle. This data would otherwise exist only as megabytes of digital data or as stacks of printouts. As you view the photographic images attached, remember the actual size of the graphic is 2' by 3'. See examples at Figures 1, 2 and 3. See if you can find the case of fratricide.*

The graphic was produced on an HP7576 plotter. A 486/16Mb RAM/66Mhz PC ran SAS 6.02. To understand the output graphic, one must also understand the input data and how it was collected.

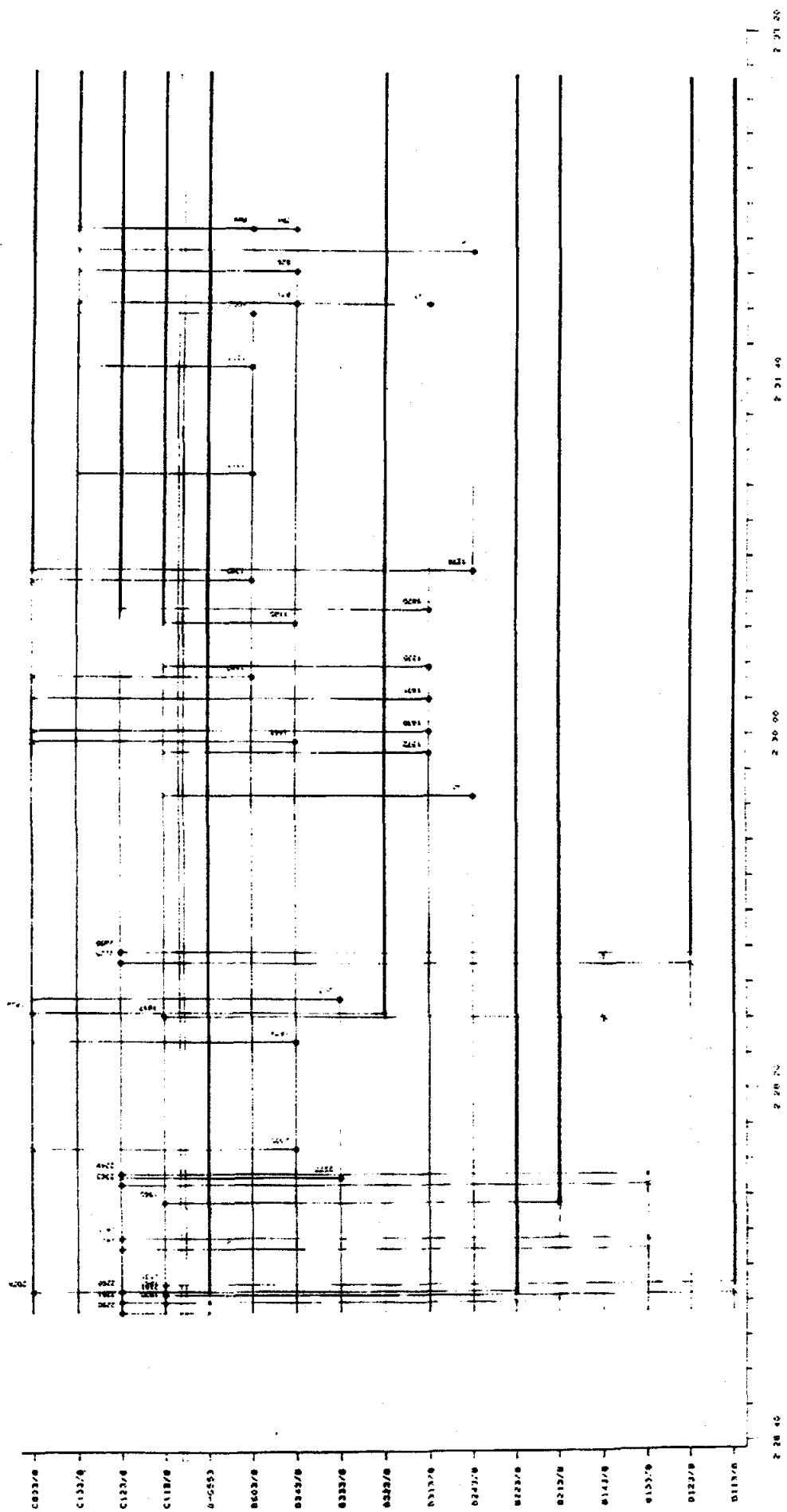
*The tank battle data was collected using a unique instrumentation system designed and built for especially for TEXCOM. Each armed player vehicle carries laser transmitters and receivers. Each also carries a small computer loaded with engagement probability tables. Each engagement is scored by the type of weapon system against the target over the engagement distance. If a player vehicle is assessed as destroyed by instrumentation, a smoke marker is set off and the player is digitally removed from the battle. At the conclusion of the battle, the engagement data is available for automatic reduction and display. Earlier we saw video footage from outside the tank. Let's now take a look at an instrumented engagement from inside the tank. Note the flashes of light are the guns firing and the smoke shows that a player vehicle is destroyed. *** 2 minute video of battle from inside a tank***.*

We have talked about the battlefield of the 21st century, the Abrams M1A2 tanks, and TEXCOM. We have also seen how SAS/GRAPH/ANNOTATE was used to build battle attrition timeline charts "The Tank Battle of the 21st Century at TEXCOM".

M1A2 IOTE

FORCE -- ON -- FORCE

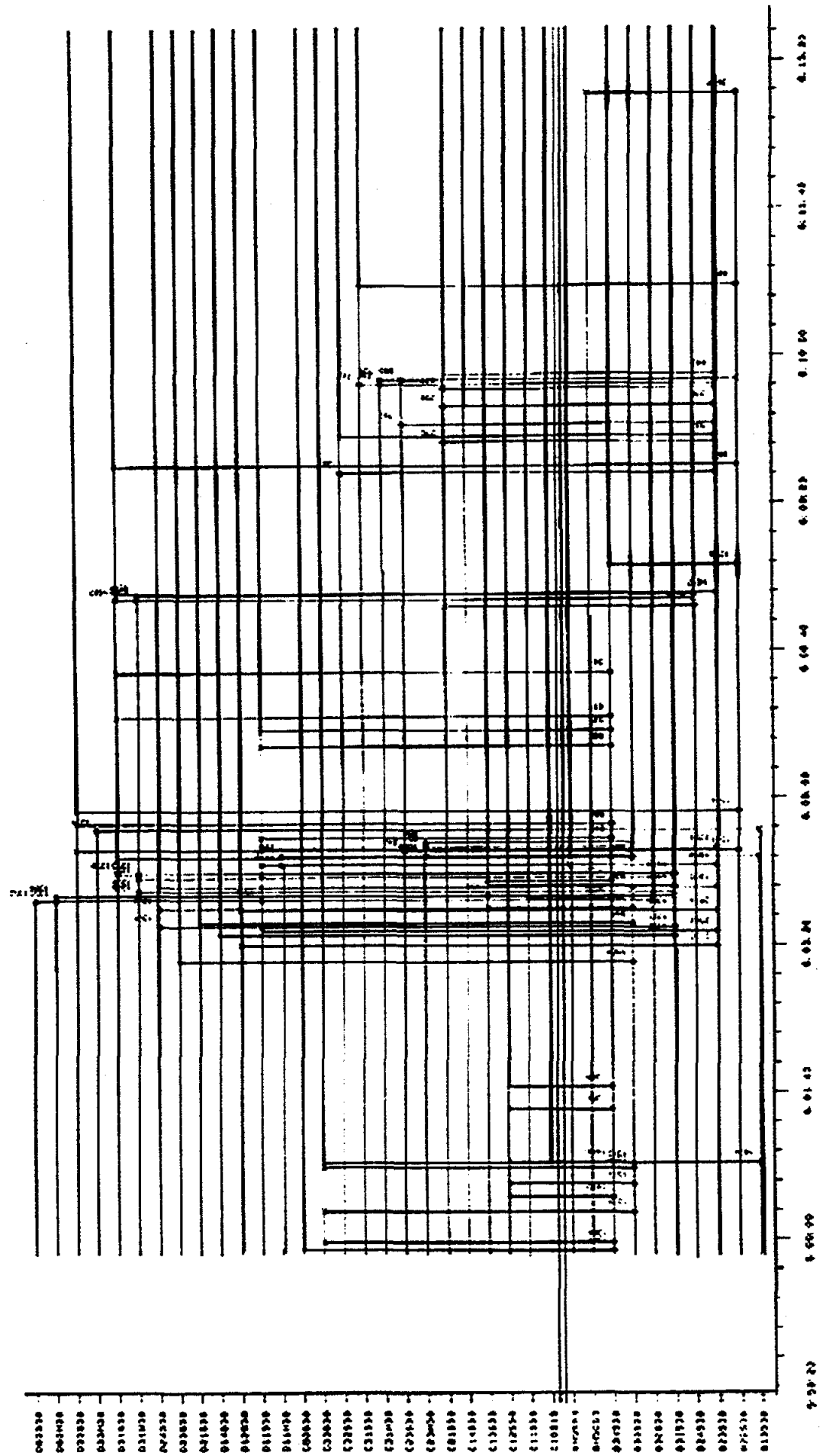
SCENARIO -- F130401A

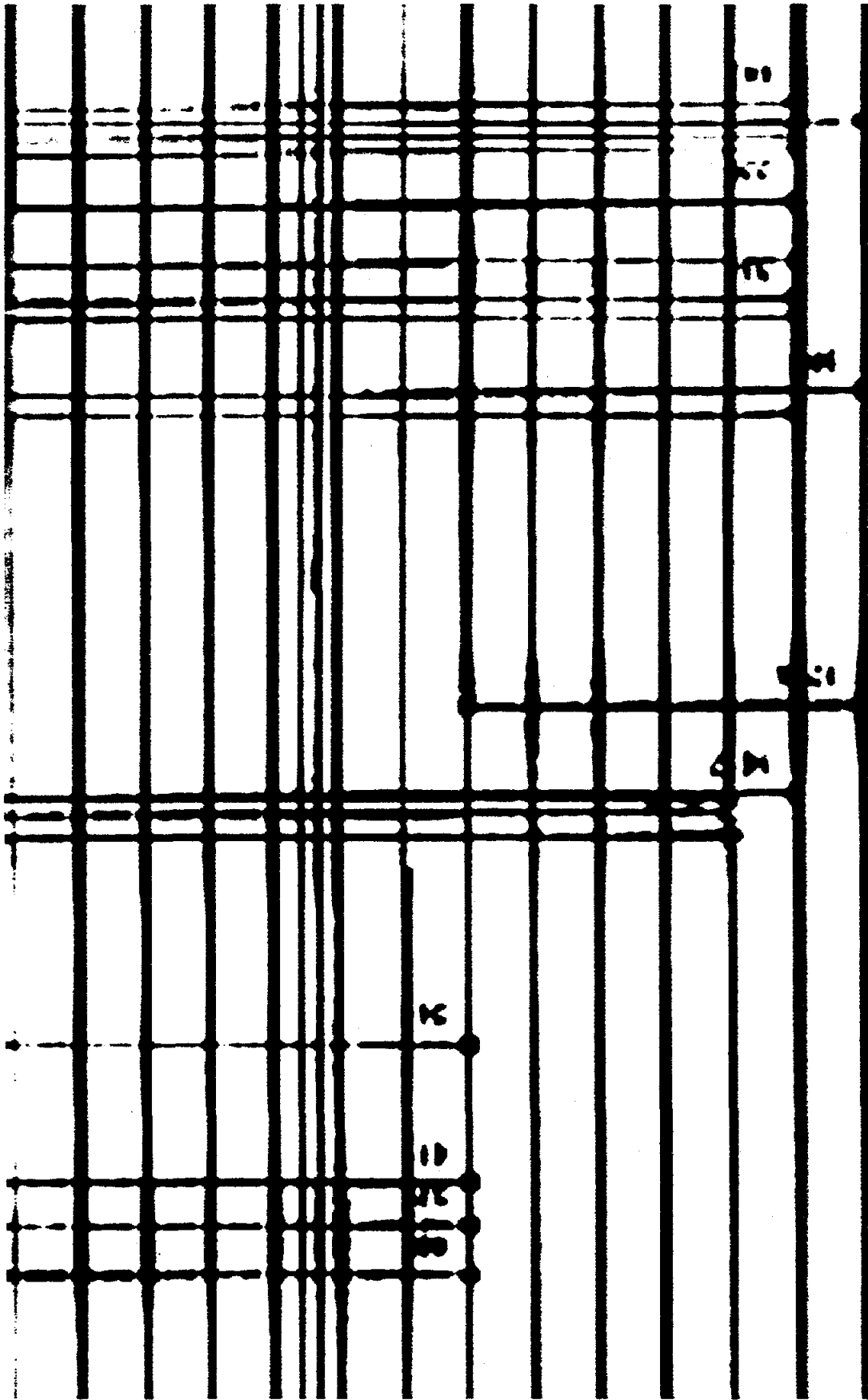


M1A2 IOTE

FORCE-ON-FORCE

SCENARIO - P131502E





30 0:00:00 0:10:1

The input data structure from a SAS prospective:

Force-on-Force Battle F131501E 15:27 Monday, April 1, 1996
CONTENTS PROCEDURE

Data Set Name: WORK.F131501E

Observations: 704

Member Type: DATA

Variables: 7

-----Engine/Host Dependent Information-----

Data Set Page Size: 15872

Number of Data Set Pages: 29

First Data Page: 1

Max Obs per Page: 25

Obs in First Data Page: 23

-----Alphabetic List of Variables and Attributes-----

#	Variable	Type	Len	Pos	Format	Description
5	F_BUMP	Char	200	32		ID of tank firing
2	F_TIME	Num	8	8	TIME8.	Time tank fired
6	H_BUMP	Char	200	232		ID of tank hit
3	H_TIME	Num	8	16	TIME8.	Time tank hit
7	K_BUMP	Char	200	432		ID of tank killed
4	K_TIME	Num	8	24	TIME8.	Time tank killed
1	RANGE	Num	8	0		Distance of engagement

The example of input data:

Force-on-Force Battle F131501E 15:08 Monday, April 1, 1996

OB RANGE F_TIME H_TIME K_TIME F_BUMP H_BUMP K_BUMP

90 .	19:56:19	.	.	B11S38	.	.	Fire
91 .	19:56:28	.	.	B11S38	.	.	Fire again
92 1897.67	19:56:32	19:56:32	.	B11S38	D12S19	.	Fire & hit
93 1983.31	19:56:32	19:56:32	.	B11S38	D21S19	.	Fire & hit
94 .	19:56:49	.	.	B11S38	.	.	Fire
95 .	19:57:16	.	.	B11S38	.	.	Fire
96 1771.95	19:57:19	19:57:19	.	B11S38	D21S38	.	Fire & hit
97 1685.09	19:57:20	19:57:20	.	B11S38	D23S19	.	Fire & hit
98 .	.	.	19:57:37	.	.	D23S19	Kill

The SAS code:

The SAS code appears here as it was last used. The heart of the program is the Annotate Data Step. All other steps set up the input data. The procedure was set to switch between the screen and the laserjet printer to ensure accuracy before going to the large plotter.

```
*****
*** PROGRAM NAME: FIREPLOT.SAS ***
*** WRITTEN BY: RAY THOMPSON ***
*** PURPOSE: Plot tank battle over time ***
*****;
***GOPTIONS: DEVICE=HPLJ150 - HP Laserjet Printer ***
*** DEVICE=HP7576 - HP Large Drum Plotter ***
*** DEVICE=VGA - Monitor Screen ***
*****
*GOPTIONS DEVICE=HPLJ150 gprotocol=sasgstd gaccess=sasgstd
gsfname=graph gsfmode=replace gsflen=80;
OPTIONS PS=58 LS=080;
GOPTIONS reset=all gunit=pct /* ftext=swissb htitle=3 htext=2 */
HSIZE = 32.693 IN VSIZE = 21.638 IN
device=VGA; * print graph to screen *;
/* device=HP7576; */
*****
***Input data from instrumentation on tanks ***
*****
LIBNAME LIB 'T:\FOFPLOT';
DATA A;
SET LIB.F131501E;
*****
*** Adjust to eliminate invalid times at the beginning or end of the battle ***
*****;
* IF F_TIME > . AND F_TIME < '20:09:00'T THEN DELETE;
RUN;
*****
***DATA MIN_MAX: Data step sets the graphic boundries on the minimum***
***and maximum time. The minimum time is the first time anything happens.**
***The maximum time is the last time anything happens. Please note that at***
```

***the end of file 45 seconds is added to the maximum time to keep the last ***
 ***engagement off the right edge of the plot. In our example the battle ***
 ***started at 8:09 PM or 20:09:00. Some other limits are checked to ***
 ***eliminate instrumentation transients in time and location. ***

DATA MIN_MAX;
 LENGTH F_BUMP H_BUMP K_BUMP \$ 6;
 SET A END=EOF;

***F_BUMP: The bumper number (ID) of the firing tank. ***

***H_BUMP: The bumper number (ID) of the tank hit. ***

***K_BUMP: The bumper number (ID) of the tank killed. ***

F_BUMP = TRIM(LEFT(F_BUMP));

H_BUMP = TRIM(LEFT(H_BUMP));

K_BUMP = TRIM(LEFT(K_BUMP));

KEEP MIN_T MAX_T;

RETAIN MIN_T '23:59:59'T MAX_T '00:00:00'T;

IF ((RANGE < 700000 AND F_BUMP > ' ') AND (H_BUMP > ' ' OR K_BUMP > ' '))

THEN

DO;

IF F_TIME > MAX_T THEN MAX_T = F_TIME;

IF H_TIME > MAX_T THEN MAX_T = H_TIME;

IF K_TIME > MAX_T THEN MAX_T = K_TIME;

IF (F_TIME > . AND F_TIME < MIN_T) THEN MIN_T = F_TIME;

IF (H_TIME > . AND H_TIME < MIN_T) THEN MIN_T = H_TIME;

IF (K_TIME > . AND K_TIME < MIN_T) THEN MIN_T = K_TIME;

END;

FORMAT MIN_T MAX_T TIME8.;

IF EOF THEN

DO;

MAX_T = MAX_T + '00:00:45'T;

OUTPUT;

END;

RUN;

***DATA HIT KILL: This data step modified bumper number (ID). This ***

****compensated for the action of instrumented control (T) or Headquarters* ***
 ****(H) vehicles. B in the bumper number indicated the friendly force or* ***
 ****blue.* ***

DATA HIT KILL;

LENGTH F_BUMP H_BUMP K_BUMP BF_BUMP RF_BUMP \$ 6.;

IF N_ = 1 THEN SET MIN_MAX;

SET A;

****Eliminate ranges over 700000. These are transient sun reflections.* ***

****Also, reduce variable length.* ***

IF RANGE > 700000 THEN DELETE;

F_BUMP = TRIM(LEFT(F_BUMP));

H_BUMP = TRIM(LEFT(H_BUMP));

K_BUMP = TRIM(LEFT(K_BUMP));

FORMAT RANGE 6.;

RETAIN MY_RANGE 0;

IF H_BUMP > ' ' THEN MY_RANGE = RANGE;

IF K_BUMP > ' ' THEN

DO;

RANGE = MY_RANGE;

MY_RANGE = 0;

END;

DROP MY_RANGE;

ONE = 'B';

IF F_BUMP =:'H' OR F_BUMP =:'T' THEN

DO;

TWO = SUBSTR(F_BUMP,1,5);

F_BUMP = ONE || TWO;

END;

IF H_BUMP =:'H' OR H_BUMP =:'T' THEN

DO;

TWO = SUBSTR(H_BUMP,1,5);

H_BUMP = ONE || TWO;

END;

IF K_BUMP =:'H' OR K_BUMP =:'T' THEN

DO;

```

    TWO = SUBSTR(K_BUMP,1,5);
    K_BUMP = ONE || TWO;
END;
IF (F_BUMP =: 'B' OR F_BUMP =: 'H' OR F_BUMP =: 'T')
THEN BF_BUMP = F_BUMP;
ELSE                                RF_BUMP = F_BUMP;
    IF (F_BUMP > ' ' AND H_BUMP > ' ') THEN OUTPUT HIT;
    IF (F_BUMP > ' ' AND K_BUMP > ' ') THEN OUTPUT KILL;
RUN;
PROC SORT DATA = HIT; BY F_BUMP RANGE F_TIME ; RUN;
PROC SORT DATA = KILL; BY F_BUMP RANGE F_TIME ; RUN;
DATA HITKILL;
    MERGE HIT KILL;
    BY F_BUMP RANGE;
RUN;
PROC SORT DATA = HITKILL; BY F_BUMP H_BUMP F_TIME;
DATA FIREPLOT;
    SET HITKILL;
    BY F_BUMP H_BUMP;
RETAIN KTR 0 HOLD '00:00:00'T;
FORMAT HOLD TIME8.;
IF FIRST.F_BUM0P OR FIRST.H_BUMP THEN
DO;
    HOLD = F_TIME;
    KTR = 0;
    END;
KTR + 1;
IF (KTR > 1 AND (SUBSTR(F_BUMP,4,1) = 'M'
    OR SUBSTR(F_BUMP,5,1) = 'M'
    OR SUBSTR(F_BUMP,6,1) = 'M')) THEN
DO;
*****
*** Instrumentation has a 3 second poling cycle and requires a fire for a hit. ***
*****
    IF F_TIME LE (HOLD + ((KTR - 1) * 3)) THEN RETURN;
    ELSE
        DO;
            HOLD = F_TIME;

```

```

        KTR = 1;
    END;
END;
OUTPUT;
RUN;
PROC PRINT; * DATA = A;
    VAR BF_BUMP RF_BUMP F_BUMP H_BUMP K_BUMP RANGE F_TIME
    MIN_T MAX_T;
    TITLE1 H=5 F=SWISSB 'M1A2 IOTE';
    TITLE2 H=3 F=SWISSB 'FORCE-ON-FORCE';
    TITLE3 H=2 F=SWISSB "SCENARIO - F131501E";
run;
*****
***DATA ANNO:    Data step sets up the plot.                ***
*****
DATA ANNO;
    LENGTH FUNCTION STYLE COLOR $ TEXT $ 20;
    RETAIN XSYS '2' YSYS '2' HSYS '1' WHEN 'a';
    SET FIREPLOT;
    IF K_BUMP > ' ' THEN DO;
        /* create observation for drawing black horizontal lines from kill to end */
        function = 'move'; x=max_t; yc=k_bump; output;
        function = 'draw'; color = 'black'; style = 'swissb'; x=k_time; yc=k_bump;
        line = 1; size = 20;
        output;
    END;
    /* create observation for brown horizontal lines from begin to fire */
    function = 'move'; x=min_t; yc=f_bump; output;
    function = 'draw'; color = 'brown'; size = 15; text = '_'; position = '9';
    style = 'greek'; x=f_time; yc=f_bump; line = 1; size = 1;
    output;
    IF K_BUMP > ' ' THEN DO;
        /* create observation for brown horizontal lines from begin to kill */
        function = 'move'; x=min_t; yc=k_bump; output;
        function = 'draw'; color = 'brown'; size = 15; text = '_'; position = '9';
        style = 'greek'; x=k_time; yc=k_bump; line = 1; size = 1;
        output;
    END;

```

```

IF H_BUMP > ' ' THEN DO;
  /* create observation for brown horizontal lines from begin to hit */
  function = 'move'; x=min_t; yc=h_bump; output;
  function = 'draw'; color = 'brown'; size = 15; text = '_'; position = '9';
  style = 'greek'; x=h_time; yc=h_bump; line = 1; size = 1;
  output;
END;
IF RF_BUMP > ' ' THEN DO;
  /* create the observation for drawing the red line from fire to hit */
  function = 'move'; x=f_time; yc=rf_bump; output;
  function = 'draw'; color = 'red'; size = 15; text = '_'; position = '9';
  style = 'greek'; x=h_time; yc=h_bump; line = 1; size = 1;
  output;
END;

IF BF_BUMP > ' ' THEN DO;
  /* create the observation for drawing the blue line from fire to hit */
  function = 'move'; x=f_time; yc=bf_bump; output;
  function = 'draw'; color = 'blue'; size = 15; text = '_'; position = '9';
  style = 'greek'; x=h_time; yc=h_bump; line = 1; size = 1;
  output;
END;
IF RF_BUMP > ' ' THEN DO;
  /* create the observation for drawing the red line from fire to kill */
  function = 'move'; x=f_time; yc=rf_bump; output;
  function = 'draw'; color = 'red'; size = 15; text = '_'; position = '9';
  style = 'greek'; x=k_time; yc=k_bump; line = 1; size = 1;
  output;
END;
IF BF_BUMP > ' ' THEN DO;
  /* create the observation for drawing the blue line from fire to kill */
  function = 'move'; x=f_time; yc=bf_bump; output;
  function = 'draw'; color = 'blue'; size = 15; text = '_'; position = '9';
  style = 'greek'; x=k_time; yc=k_bump; line = 1; size = 1;
  output;
END;
/* create the observation for writing range on fire line */
  function = 'label'; color = 'green'; position = '9'; style = 'simplex';

```

```

size = 1; text = ' ||left(put(range,6.)); x=f_time; yc=f_bump; ANGLE = 90;
output;
run;
*****
***Finally! Plot!***
*****
PROC GPLOT DATA = FIREPLOT;
PLOT F_BUMP*F_TIME=1 BF_BUMP*F_TIME=1 RF_BUMP*F_TIME=1
H_BUMP*H_TIME=2 K_BUMP*K_TIME=3 K_BUMP*MIN_T=4
K_BUMP*MAX_T=4 F_BUMP*MIN_T=4 F_BUMP*MAX_T=4
H_BUMP*MIN_T=4 H_BUMP*MAX_T=4
/OVERLAY
annotate=anno;
SYMBOL1 V=+ C=BLACK; SYMBOL2 V=X C=RED;
SYMBOL3 V=STAR C=GREEN; SYMBOL4 V=NONE; * C=CYAN;
format f_time h_time k_time min_t max_t time8.;
FOOTNOTE1 A=90 ' ';
LABEL F_TIME = 'TIME' F_BUMP = 'BUMPER NUMBER';
TITLE1 A=90 ' ';
TITLE2 H=5 F=SWISSB 'M1A2 IOTE';
TITLE3 H=3 F=SWISSB 'FORCE-ON-FORCE';
TITLE4 H=2 F=SWISSB "SCENARIO - F131501E";
RUN;

```