YESTERDAY'S NEWS

VOLUME 4 NUMBER 11 Established 2016 NOVEMBER 2019

30 Years Ago...

Historical Information taken from Bill Gaskills TIMELINE

NOVEMBER 1989:

TRIS, a Jim Reiss take-off of the TETRIS game written by a Russian, is released by Asgard Software.

Gene Hitz, of Arcade Action Software, releases Touchdown 89, Snowmobile Derby, Moon Run and Cut Throat Cribbage.

Comprodine releases Color Card and Color Flyer graphics programs for the XR-1000 printer, Artist Print Shop by Paul Coleman and Living Tomb and War Zone games by Quinton Tormannen.

Texaments releases Mailing List Manager for TI-Base, written by Bill GasKill.

Asgard Software releases Karate Challenge and Mission Destruct! games by Glenn SchoraK.

Sports Pics, a companion disK for PAGE PRO 99 or TI-ARTIST written by Paul Coleman, is released by Asgard Software.

Texaments assumes exclusive distribution of Barry Boone's Exec program for the Geneve.

Myarc's Lou Phillips reports that John Birdwell has begun worK on the tape streamer backup software for use with the HFDC card.

Asgard Software releases Doom Games III by Ken Gilliland, the company's 100th pr oduct.

Asgard Software begins offering NEW module-based software for the 99/4A when it announces the release of Typewriter 99 and Tris in module versions.

The Chicago TI-Faire takes place on November 4th in Rolling Meadows, IL. It is followed on November 5th by the MilwauKee TI-Faire. YN

INSIDE TI99ERS INFORMATION

DOW 4 GAZELLE FLIGHT SIMULATOR

by
John T. Dow

BY DAVID WELCKER - 99'ER - JANUARY 1983 - VOL. 2, NO. 3

Learn to Fly! That is the challenge of Gazelle by John T. Dow. From the first roar of the Gazelle's engine, this real-time simulation will Keep you on the edge of your cocKpit seat. And it does so without a single laser blast or invading alien! What grabs your attention is your sudden take-off as the pilot of a single-engine plane and the scenario is liKely to include a frenzied operator, a dog-eared manual, and joysticKs wet with perspiration.

Before taKing control, you have to absorb thirty pages of text and several drawings, and you can expect to consult the manual for at least the first few flights. During flight, the screen displays a dashboard - ten dials and eleven indicator lights which the student pilot must understand and manipulate via the joystick and Keyboard controls. Above the dash is an indicator in the night sKy showing the distance and direction to the landing field.

See "DOW4", Page 2

TI CLASSROOM

TIPS FROM THE TIGERCUB

NUMBER 6

By Jim Peterson

```
100 CALL CLEAR
110 PRINT "  THE TIGERCUB'S
CHRISTMAS": :TAB(11);"ORNAME
NT": :"by Jim Peterson": :
120 PRINT " Use arrow Keys a
nd W,R,Z":"and C Keys to mov
e cursor,":"Then press any O
THER Key to"
130 PRINT "print design, or
Enter to":"sKip. Use both sh
ifted and":"unshifted letter
s and punct-"
140 PRINT "uation, also type
 any letter":"with CTRL held
down."
150 PRINT " Use FCTN 1 to ch
ange pat-":"tern of last cha
racter prin-":"ted, or FCTN
2 to change its"
160 PRINT "color, or FCTN 3
to change":"all the colors,
or FCTN 7":"to slowly change
 all the"
170 PRINT "patterns - but st
ay away":"from FCTN 4 and FC
nt = !!":"       Press any
Key"
180 CALL KEY(0,K,ST)
190 IF ST=0 THEN 180
200 DIM A$(16)
210 CALL CLEAR
220 FOR SET=1 TO 16
230 CALL COLOR(SET,5,16)
240 NEXT SET
250 CALL VCHAR(1,31,1,96)
260 CALL SCREEN(2)
270 FOR CH=33 TO 159
280 PRINT CHR$(CH);" ";
290 Z=Z+1
300 IF Z/14<>INT(Z/14)THEN 3
20
310 PRINT : :
320 NEXT CH
330 GOSUB 630
340 GOSUB 940
350 CALL VCHAR(1,3,32,672)
360 CALL VCHAR(1,29,1,192)
370 R=6
380 C=11
390 CALL KEY(3,K,ST)
400 IF ST=0 THEN 390
410 IF K<8 THEN 820
420 ON POS("WERDCXZS",CHR$(K
),1)+1 GOTO 390,490,470,460,
440,430,530,520,500
430 R=R+ABS(R<12)
440 C=C+ABS(C<16)
450 GOTO 540
460 C=C+ABS(C<16)
470 R=R-ABS(R>1)
480 GOTO 540
490 R=R-ABS(R>1)
500 C=C-ABS(C>5)
510 GOTO 540
520 C=C-ABS(C>5)
530 R=R+ABS(R<12)
540 CALL HCHAR(R,C,42)
550 CALL KEY(5,CH,ST)
560 IF ST=0 THEN 550
570 IF (CH=87)+(CH=69)+(CH=8
2)+(CH=83)+(CH=68)+(CH=90)+(
CH=88)+(CH=67)THEN 550
580 CALL HCHAR(R,C,CH)
590 CALL HCHAR(25-R,C,CH)
600 CALL HCHAR(R,33-C,CH)
610 CALL HCHAR(25-R,33-C,CH)
620 GOTO 390
630 DATA 00,18,24,3C,42,5A,6
6,7E,81,99,A5,BD,C3,DB,E7,FF
640 FOR J=1 TO 16
650 READ A$(J)
660 NEXT J
670 FOR CH=33 TO 159
680 GOSUB 700
690 GOTO 800
700 FOR L=1 TO 4
710 RANDOMIZE
720 X=INT(16*RND+1)
730 B$=B$&A$(X)
740 C$=A$(X)&C$
750 NEXT L
760 CALL CHAR(CH,B$&C$)
770 B$=NUL$
780 C$=NUL$
790 RETURN
800 NEXT CH
810 RETURN
820 ON K+1 GOTO 390,830,390,
850,830,390,390,820
830 GOSUB 670
840 GOTO 390
850 IF (CH<33)+(CH>159)THEN
390
860 GOSUB 700
870 GOTO 390
880 SET=INT(CH/8)-3
890 IF SET<2 THEN 390
900 GOSUB 980
910 GOTO 390
920 GOSUB 940
930 GOTO 390
940 FOR SET=2 TO 16
950 RANDOMIZE
960 GOSUB 980
970 GOTO 1030
980 X=INT(15*RND+2)
990 Y=INT(15*RND+2)
1000 IF Y=X THEN 990
1010 CALL COLOR(SET,X,Y)
1020 RETURN
1030 NEXT SET
1040 RETURN
```

This routine contains many ideas which may be educational to beginning programming. Lines 110-170 will show how to neatly format the text on the screen.Lines 180-190 hold the text on the screen until any Key is pressed; this is much better than a delay loop, because different people read at much different speeds. Lines 220-260 create a white screen with a blacK border, which maKes colored graphics stand out much more brightly After a CALL CLEAR, the screen is actually filled, except for a strip at top and bottom, with 24 rows of 32 columns of the blanK space-bar character. The normal screen color is light green (4) in Basic or cyan (8) in Ectended Basic, and the normal color of all the character sets is blacK (2) on transparent (1); The blanK character has no foreground color and the transparent backGround color let's the screen show through. Line 230 changes all the sets to darK blue on white, so the blanK space character becomes white. The strip at the top and bottom of the screen is filled with the edge character, ASCII 31 which is transparent. Line 250 prints two columns of the edge character at the right edge, then wraps around to print two more columns at the left edge. The blacK screen color, called in Line 260, can now show through the transparent edge character on all four sides.

Lines 270-280 put all the printable characters on the screen, to givbe you something to watch while the program is being initialized The " " in Line 280 puts a blanK space after each character, and the semicolon Keeps the print on the same line until it is full. Since 14 characters can be double-spaced on a 28-column print line, a print row is sKipped whenever the value of Z becomes divisible by 14.

Now the program goes to the routine in Lines 630-810 to redefine every Keyboard character to a symmetrical ornamental pattern. The DATA statement in Line 630 contains 16 pairs of hexadecimal numbers which are mirroe images of each other; LooK at the chart on page109 of your "Beginner's Basic" to see what I mean. Lines 640-660 read these into 16 subscripts of A$; thus the dimension statement in Line 200 was necessary.

Next, Lines 670--800 randomly combine these pairs to form symmetrical characters.

Lines 680,690 and 790 are only so that we can later re-enter the middle of this routine. Lines700-750 loop 4 times to pick a random value of 1 to 16 for X, to randomly select those mirrored pairs. B$ is built up to 4 pairs of these numbers, forming the upper half of a character which will have left/right symmetry. At the same time, C$ is built up of these same 4 pairs in reverse. Finally, Line 760 combines B$ and C$ to redefine a character which is also symmetrical top/bottom Since B$ and C$ were built up by adding onto themselves Lines 770-780 must cancel them out before they are used again in the next loop of CH. NUL$ is not a magic word, it is just a string variable name which is never given a value, therefore represents an erased value.

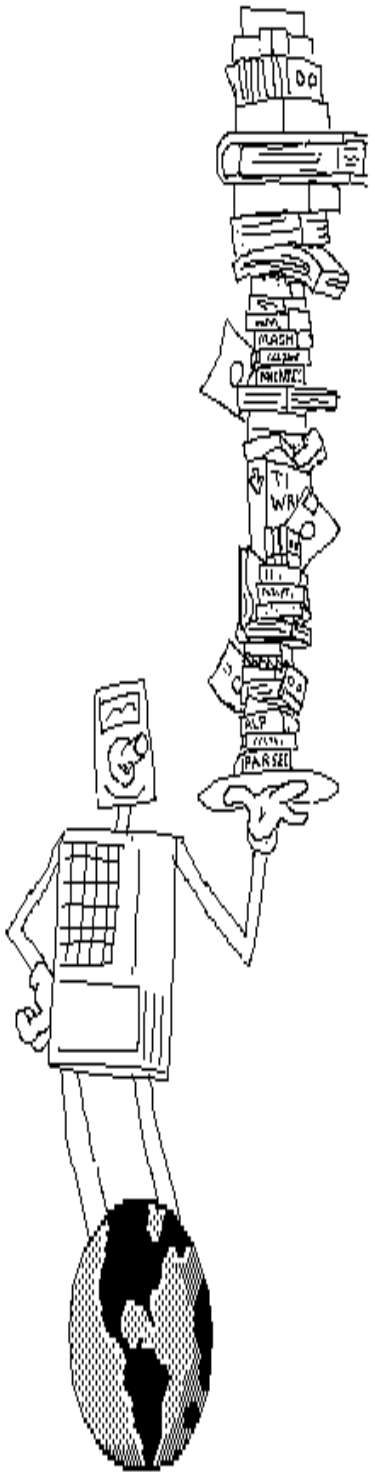
The program returns to Line 340 and then to 940, where each character set from 2 to 16 is given a different color combination selected randomly. Line 1000 insures that the foreground and background color will not be the same. Again, Lines 960-970 and 1020 are just so that we can later re-enter this routine in the middle.

Since a CALL CLEAR would erase our screen borders of ASCII 1, the screen is cleared by the CALL VCHAR wipe in Line 350, and Line 360 then widens this transparent border to leave only a 24x24 square.

Lines 390-540 are a very compact and efficient routine to move the cursor with the arrow Keys and diagonal Keys, without going beyond predetermined borders.

OUT OF MEMORY, HAPPY HACKIN'

Jim Peterson



DOW4 continues...

As this is the only noninstrument indicator, the simulation is even more difficult than daytime flying - you can't even look out the window to see how the plane is situated.

When you finally land the plane (under these circumstances it took me about fifteen tries) there is the further challenge of instrument landings with more sensitive controls, short-field landings, and rapid take-offs as well as challenges for the advanced pilot, suggested in the manual. After mastering them, the pilot can choose to tackle maneuvers under varying turbulence. As a result, the program keeps its appeal, even when mastered. And unlike some challenges for 99'ers, the expert sessions need not become marathon.

Speed of response to your controls is excellent and realistic. A time out option lets you freeze the action in order to deal with backseat drivers, calls of nature, and disgruntled spouses. If you make a mistake (by flying too fast, missing the runway, or losing control), you'll hear the crash followed by the whine of the ambulance. The cause of your misfortune is displayed with other flight data, including start options. Safe landings earn a musical pat on the back and the warm satisfying glow that comes only with mastery of this demanding task.

Some will complain that Gazelle's instruments are hard to read, in that pixels (dots) make up the indicator needles and calibrations. I have not found this hard to get used to, but if you have eyeglasses, be sure to wear them. In any case, the increased speed is worth the squinting. The program makes use of the full 16K, thus some of the usual title screen/text amenities are left out (and barely missed).

And best of all, you really can learn the principles of flying! The interaction of the controls and indicators is quite well integrated. The game is so realistic, that the level of understanding needed to play it probably excludes anyone too young to fly a real plane in the first place!

Whether to teach or to entertain, this simulation will meet all your expectations. And should you accept the challenge of Gazelle, please watch for me up there my registration is Tango India Niner Niner Four Alpha.

Dow4 Gazelle by John T. Dow is available in TI BASIC (joysticks required) for $30.00 on cassette. YN

By John Kolean - MICROpendium
August 1985 - Vol. 2. No.7

REPORT CARD	
PERFORMANCE	A
EASE OF USE	B
DOCUMENTATION	B+
VALUE	B+
FINAL GRADE	B+

The Model Dow-4 Gazelle has been around some three years and remains as one of only two flight simulator programs available for the TI. The other is Winging It by Not Polyoptics. Both

of these programs run out of console BASIC.

The Model Dow-4 is an entertaining and challenging simulation for adults and teens. It is far too complex for children.

To be perfectly honest, I did not expect a great deal from the program, considering it is programmed to run with no more than a TI console, and am surprised by how well it maintained my interest.

This is not to say that the program is as good as it can get. I am even more surprised that Mr. Dow hasn't written an Extended Basic, disK-based version that taKes better advantage of what the TI has to offer. I will have more to say about this later.

Performance: The Dow-4 Gazelle is a four-place, single-engine airplane. The craft has a maximum speed of 200 miles per hour, a service ceiling of 16,400 feet and a panel of 15 instruments that report everything from airspeed to the position of the flaps. Each is updated about once every second, providing an array of constantly changing displays.

Input is based on the joystick and the Keyboard. The joystick serves as the yoKe and gives the user an easy-to-grasp feel for how the plane behaves. Keyboard inp ut is used to control the throttle, fuel tanKs, flaps, instrument navigation, pitch, joystick pitch effectiveness and automatic wing leveling. There is also a stall warning indicator and a pause Key, both of which come in very handy.

Modes of operation include: TaKe Off, In Flight, Landing, Short Field Landing and ILS (Instrument Landing System) approach. While selecting the mode, the user also determines the extent of wind and turbulence to be encountered.

The TaKe Off mode is self-explanatory while the In Flight mode allows the user to specify the starting velocity, the starting altitude and the ground level. The Landing mode places the user in the landing pattern at an altitude of 1,000 feet and a velocity of 130 mph. The Short Field Landing is similar to the Landing mode except that the field is shorter and there is a 400-foot high ridge one-half mile east of the runway. (Not being able to see the ridge increases the difficulty of this maneuver. The JLS Approach is exceedingly difficult and requires a great deal of attention to detail to be successful at it.

I found the instruments and controls to perform in a realistic manner. Everything worKs in synchronicity. The pitch, position of the yoKe, amount of power, position of the flaps all contribute to the realism of the simulation.

The instrument panel itself is realistic, though some instruments are difficult to read, most of which is due to the fact that the instruments themselves are relatively small, allowing a considerable amount of space between each. The instruments could easily have been made larger, and thus easier to read.

Although there is only one airport, with landings made from approaches at 90 or 270 degrees, I found the act of flying and landing the Dow-4 Gazelle to remain a challenge. Timing is everything in this simulation and it is easy to over- or underestimate approach velocity and angle of descent. Sound effects play an important part in the simulation. The sound of the engine is well done, changing based on the based of RPM. Sound is much more effectively used in the Dow-4 Gazelle than in Flight Simulator II.

My only criticism has to do with the feeling on my part that this simulation has not been carried as far as it could have been, particularly in view of the fact that it has been marKeted since 1982. A disK-based, Extended BASIC version, using expansion memory, could add new dimensions in the performance of the simulation. The addition of overlays on disK that can be read into the computer as needed could create the ability to fly from one airport to another. It isn't necessary to be able to see out of the cocKpit ala Flight Simulator II (even with FSII the user is primarily watching instruments). But it would add to the realism of the simulation to be able to fly from point to point without running out of fuel and to land at a variety of airports each of which has its own characteristics and challenges.

Ease of Use: It taKes a while to master this program. Each flight is different, and the addition of turbulence and strong winds can maKe even the most routine flight a challenge. It tooK me several hours to be able to successfully land, but once I did I felt I had accomplished something. Following unsucessful landings the user is treated to the sound of a siren that is followed with a display that notes the wind direction and speed, rate of descent, cloud ceiling, G force, speed, altitude above sea level and the heading. This information is also displayed following a successful landing but without the siren. Also, the simulation reports the causes of any crashes, ranging from missing the runway to losing the plane's wings as the result of pulling excessive Gs.

Documentation: The Dow-4 Gazelle comes with a 38-page manual. It includes a table of contents and glossary and backGround information about flying. There are also several drawings, depicting the instrument panel with explanations of each instrument and two examples of approaches to an airport with hints about how to land successfully.

Value: This program is well done as far as it goes, but the fact that it is done in console BASIC and is available only on cassette limits its potential in the mature TI marKetplace. The use of sprite graphics could do a great deal to enhance the readability and appearance of the control panel. The use of memory expansion would have allowed a fuller development of the simulation. The reliance on instrumentation is not a handicap.

Overall, this program remains a challenge, and is worth the price. I only wish that there was a Dow-4 Gazelle II available that really taKes advantage of the TI (in Extended BASIC with 32K). YN

WINGING IT

Text from the Not Polyoptics catalog

Fly your own private plane in this excellently visualized simulator. But after you've mastered flying, the excitement isn't over - three different games test your sKill. Ground map and weather indicator chart your position on a randomized earth simulacrum. For 1 player. *"an excellent program, which should appeal to budding pilots of all ages". Home Computing WeeKly.*

Text from the WINGING IT manual

You gun the 125 hp Menasco C-4 engine and your Arrow monoplane begins to move slowly along the runway. The plane taxies faster and faster, and the air is rushing into the open cockpit. 30 knots, 60 knots, 90 knots - carefully you pull back on the stick and your stomach seems to drop. The horizon indicator sinks, and the altitude gauge reads 10 feet. You're flying!

Winging It is Not Polyoptics inexpensive flight simulator written for the unexpanded TI 99/4A computer. It was written in TI Basic, which at times is a slow and inefficient language. Two problems result result, the execution of the program is rather slow; and, second, the computer will only recognize Key input at certain points in the program. In order to minimize the effect of the first problem, the simulation has been designed to compensate by recalculating the plane's position at the several second intervals that the display taKes to be updated. Thus, a "real time" experience is preserved. The second problem means that a command Key must be held down until it taKes effect, generally speaKing, about one-half the recalculation cycle. A change in the engine noise will usually mean that the program has passed through a Key input recognition statement. Winging It may be played in Extended Basic to enhance its performance.

MICROPENDIUM
May 1986
Volume 3
Number 4

By John Koloen

REPORT CARD	
PERFORMANCE	C-
EASE OF USE	A
DOCUMENTATION	A
VALUE	B
FINAL GRADE	B-

My first reaction to 4A Flyer was delight. Scanning the most recent Triton Products catalog my eyes seized on the blurb for 4A Flyer. But after receiving the cartridge-based program and trying it out, my initial enthusiasm turned sour. I became disappointed. After a couple of minutes at the console it became apparent that this is not the serious simulation of flying that I had hoped far. Rather, it is a relatively superficial program. Instead of teaching one something about flight - which any good flight simulator should - even those who have never flown must suspend their credulity to overcome some of its most apparent deficiencies. Chief among these is the presence of certain aerodynamic impossibilities, which leads me to question whether it should be described as a flight simulator, as its manual states.

Prior to receiving 4A Flyer I had hopes that it would build on what John Dow started with his Model Dow-4 Gazelle program. Written in Console BASIC, the Dow-4 simulation is realistic in its aerodynamics but limited by its graphics and the fact that it is designed to be run out of a console and a cassette recorder. 4A Flyer comes in a cartridge, is programmed in assembly language but lacKs much of the sophistication of the Dow-4 simulation.

Performance: 4A Flyer is easy to use and master. The airplane can be controlled using either the Keyboard or joysticK, the joysticK being preferred. After plugging the cartridge in, one is prompted with a weather option. Selecting this option may result in adding inclement weather to the flying experience. Clear weather is the normal condition, with snow, rain and heat possible via the weather option. One is made aware of the the type of weather condition by the color of the sKy and ground. A blue sKy and white ground, for example, means snowy weather. Of course, the various weather conditions pose hazards, such as ice forming on wings at high altitudes or a slippery runway while landing in a rain storm. (I did not have an opportunity to try out the program on a blacK and white or monochrome monitor. However, it maKes good use of color.)

Throughout the simulation the user is faced with an instrument panel that includes an altitude indicator, pitch indicator, landing gear indicator, braKe indicator, artificial horizon, fuel gauge, engine power gauge, speed indicator, heading indicator and flap indicator.

The simulation starts with the plane at rest on a runway. Engine speed is increased by pressing the "1" Key, decreased by pressing the "2" Key. "I" raises the flaps, "M" lowers them; "0" raises the landing gear, ";" lowers them; and "B" applies the braKe, "," releases it. TaKing off is simply a matter of increasing the speed of the plane as it proceeds down the runway and pulling backK on the joystickK. There is no provision for steering the plane during taKeoff or landing, another defect in the simulation.

After getting airborne, the landing gear should be raised. Then the user may go to a cruising altitude, land, or engage an enemy plane in combat. Pressing "C" initiates the combat mode, "," ends it.

A digital readout is used to Keep trackK of one's bearings. While the plane's heading is depicted on the left side of the screen, the lower center portion includes a square that contains information about the altitude of enemy planes, bearings of enemy planes and runways and the number of times one has been hit by enemy gunfire.

To engage the enemy, one presses the "C" Key, and a readout on the enemy plane's location appears. The user climbs or descends to the enemy's altitude and banKs to its heading. The enemy then appears in the cocKpit window at the top half of the screen. Both you and the enemy are on a collision course. The sprite graphics depicting the enemy plane are quite well done. Using the joystickK and fire button, the user lines up the enemy in a gunsight that appears in the cocKpit window, and then fires. Hitting the enemy is trickY. I had far more misses than hits. As the enemy plane approaches, it grows larger and then, if you haven't destroyed it, it registers a hit on you and vanishes. The location of the next enemy plane then appears and you may engage it or press the comma Key and disengage the combat mode. Although flaK occasionally appears while engaging the enemy, it does not seem to damage either plane.

If one does not go into the combat mode, there isn't much else to do except land. Landing is accomplished by first pressing the "L" Key, which results in a readout of the compass bearing of the nearest runway. (Incidentally, once you approach the runway it is not possible to abort the landing procedure.) You then descend toward the ground, reducing speed, lowering flaps and landing gear until you're a few feet off the ground. A runway then appears in the cocKpit window and you try to gently touch down.

If all this seems rather matter-of-fact it's because 4A Flyer is not a very exciting simulation. Little about it is consistent. Although the program generates a noise for the engine that approximates what an engine sounds liKe - the pitch increasing and decreasing with engine speed - the sound used to simulate gunfire in the combat mode is dreadful. The machinegun fire sounds more liKe a buzzer at the end of a basKetball game than a weapon.

Similarly, while the plane will self- destruct if you leave the landing gear down at speeds of 125 miles per hour or greater, it isn't possible to go into a headlong dive no matter what you do. I Killed the engine at 5000 feet, pointed the nose down and watched the velocity indicator actually decrease as I descended.

It is evident that many compromises had to be made to maKe 4A flyer fit in the available cartridge ROM. Even persons with only a cursory understanding of gravity would not have overlooKed some of the deficiencies noted above.

But limiting oneself to the memory available in a cartridge is a problem when you are trying to write something as sophisticated as a flight simulation. But it may be a prerequisite when trying to reach as broad a marKet of TI users as Triton. Much of its products are aimed at users who do not have PEB boxes, and for them 4A flyer may prove to be enjoyable. But for those with expanded systems, who are accustomed to longer and more complex programs than are generally available on cassette or cartridge, this program isn't liKely to measure up to the expectations created by the program's manual or the blurb in the Triton catalog. My view is that a realistic flight simulation requires an expansion memory and disK system.

Ease of Use: 4A flyer is easy to use. Although it is called a simulation, it performs more liKe an arcade game, except for the absence of scoring, (There is a readout for the number of times you are hit by enemy gunfire. However, after the third hit the plane is destroyed and you must restart.) Whatever difficulty one has in taKing off or landing is quicKly resolved by sKimming through the manual.

Documentation: The 12-page manual is thoroughly adequate to the purpose. It is written in a style that hypes the simulation, maKing one anticipate that first taKeoff. Unfortunately, the program is not equal to the hype.

Value: When I saw this program advertised in the Triton catalog I didn't waste a moment in placing an order. Even as I waited, Triton sent me a review copy. Clearly, Triton is excited by the marKet potential of this program (since the company has never sent MICROpendium any products for review in the past).

Unfortunately, I remain disappointed by it. I feel this program should not be described as a simulation, since it is obvious that it does not actually simulate the aerodynamics of flying, It is simply too superficial to be convincing. It could be more accurately described as a psuedo-simulation, or a simulation of a simulation.

I should have Known by the price that 4A flyer couldn't possibly be an improvement over the Dow-4 Gazelle. Those who liKe their simulations on the tame side may find 4A

Flyer to be diverting, but those who want the challenge of realism in their simulations will be disappointed.

To my knowledge, there are now three flight simulation programs for the TI, none of which utilizes an expanded system. My hope is that other programmers who may be laboring on a sophisticated flight simulator for the TI will continue their efforts. 4A flyer clearly is not it.YN

Programmer John Phillips responds

MICROpendium June 1986 - Volume 3, Number 5

I am the programmer of 4A Flyer, a software product you recently reviewed in the May issue. Thank you for the lengthy review. It is obvious that your software instincts are exceptional.

The review, itself, was precisely on track with the concept, functionality and essence of 4A Flyer. It is, indeed, a simulation of a simulation and was never intended to be anything else. The deficiencies and strengths of 4A Flyer were outlined in your review and I applaud your perceptions. The only factor you were lacking in the review was the birth of 4A Flyer. Please allow me to explain.

In mid-January of 1986, Triton contacted me concerning the development of a so-called flight simulator. Their only basis of comparison was the Microsoft Flight Simulator for the IBM Personal Computer, so they had a great number of "I would like" ideas for the software content. For marketing's sake, the cartridge was to be restricted to 8K of ROM code. The only RAM allowed was the 200 bytes of free CPU RAM located within the 4A.

My first reaction to this was quite negative. A lengthy design phase was then established and an outline of what exactly the software could perform was drawn up. All that is in 4A Flyer was set forth in that design. The element of realism was to be included in the software, provided that space allowed for it. This was not to be the case.

When the design was finished, I approached Triton about programming costs and scheduling. My intent was to have the software finished by late June or early July. This, unfortunately, was not Triton's idea. In order to make the spring catalog, they had to have completed, debugged code and documentation by April 1, 1986. As you can see, that left me eight weeks. I needed help.

At this point, I knew that I could not complete the software alone. I pulled in a brilliant mathematical programmer, Michael Archuleta, to complete the sections which would allow the realism of flight to be added. This entails 3-D to 2-D rotation and translation. I provided Mike with the design and he began work immediately.

Mike used the algorithms found in a book by Bruce Artwick of Sublogic to accomplish this task. It took him nearly six weeks of work to translate the algorithms to the 4A. Although they ran in his small environment, the biggest job was still ahead: making his routines fit into my code.

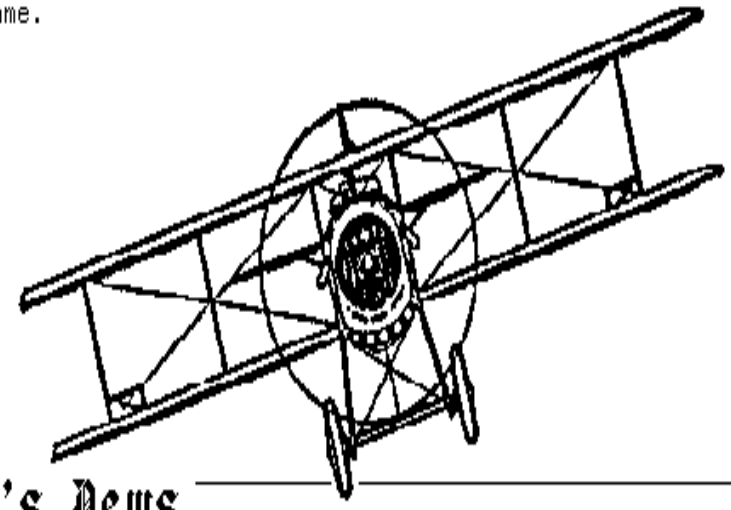
At the six weeks point, I had completed most of the graphic work, input routines, sound effects, gauge control and link routines. Mike and I worked an entire weekend adding his routines to the software. Although they had worked by themselves, several errors were uncovered which were traced directly to the algorithms in Artwick's book. In essence, Mike's work was worthless. You can probably see that there was no time to give it a second chance. The software was due in a week and a half.

At this point, I had to resort to my backup plan which, coincidentally, encompassed all the things you listed as deficiencies. In that respect, your review was as fair as could possibly be. I appreciate your honesty to the 4A users.

To make a long story short, I had eight weeks to create 8K of code. I am surprised that 4A Flyer made it out the door. It was the most difficult project I had ever worked on and the most detrimental to my sanity and home life. I can understand the sophisticated user being somewhat disappointed, but think of the thousands of users who have never pondered the essence of flying. In principle, my simulation's simulation does cover the basics and should be entertaining to novice users.

There is hope, however, for those like you who want more from this so-called dead computer. Mike and I will be working on the 32K version of the flight simulator in which we hope to add the dimensions of realism back into the software. This project should take us over one year to complete, but it will be done the way a flight simulator should be done.

Once again I thank you for your review and the consideration you gave its deficiencies and strengths. Perhaps the next revision will add credibility to its name. YN



SPAD XIII

MICROPENDIUM 03/87
Volume 4, Number2

By John Koloen

Spad XIII by Not-Polyoptics is the latest entry in the growing TI flight simulation field. It is also one of the better ones. The simulation involves a World War I motif throughout. TaKing the controls of a Spad XIII, you have the option of recreational flying or going into battle against the Huns. It's a safe bet that anyone who flies the Spad will spend most of his time in the attacK mode.

Performance: Spad XIII can be controlled using a joystick or the Keyboard. Keyboard input gives the pilot more control than a joysticK. The Keyboard became my preferred manner of flying, though it was a little awKward at first. It is often necessary to hold a Key down for a second before it registers. Occasionally this delay allowed an enemy plane to pass out of the circle of my gunsight.

The W, E, R, A, S. D, F, Z, X, C and V Keys are used to control stick movement, the comma and period Keys are used to control the rudder, the Q Key fires the machine guns, the B Key is used to drop bombs, the throttle is controlled by the 7, 8, 9 and 0 Keys, and Keys 1-6 are used to select a variety of views (front, left, right, backK, up and down). The U Key is used to provide a forward view without a cockpit. Normally, a cockpit fills half the screen. The cockpit includes an instrument panel consisting of an altimeter, compass, fuel gauge and air speed indicator. There is also a throttle indicator. The cockpit is framed by the upper wing of the Spad and the tail. The tail rises and falls depending on how level the plane is flying. The position of the tail in relation to the gunsight can be used to insure level flight, though this can be accomplished simply by eyeballing the horizon.

Spad XIII operates as you d expect. Being a rather simple airplane, not a great deal is required to get it airborne. Rev the engine, roll down the runway and pull backK on the stick is enough. Keeping it airborne takes a little practice, and it is definitely easier to do with the Keyboard than a joysticK. Operating in a WWI context, you needn't bother with obtaining clearance from a tower to land. A light plane such as the Spad can land safely in virtually any clearing. Landing on the runway takes practice, but if you miss and don't want to try again you can just set her down anywhere.

The program uses a stylized 3-D view throughout. (By stylized, I mean that objects are rendered with lines.) The effect is convincing, though not as detailed as a picture created using a good graphics design program.

The most detailed graphics in Spad XIII are the aircraft, French and German airports and hangars, German airplanes, flaK and observation balloons. There are also trenches, clouds, sun, the Eiffel Tower, villages, the Seine River, and hills. Predictably, things get bigger as you approach them and smaller as you move away from them. The closer you come the move detailed objects become. You can get close enough to an observation balloon or enemy airplane that they fill the screen.

Color is used in a basic manner: blue represents sKy and ground, sun is yellow, clouds are white, blacK are the enemy planes, red is the machine gun fire coming from your plane.

I find the graphics to be quite satisfactory.

Sound plays a role in Spad XIII, including the droning of the engine and the firing of machine guns. The engine sound changes depending on the amount of throttle. The machine gun sound is right on target. Unfortunately, flaK and bombs are not accompanied by sound.

Flying Spad requires a light touch, particularly when climbing. It is easy to angle the nose of the Spad a little too much, causing the plane to roll over. Stalling is a less common problem since it largely has to do with velocity. You expect to stall when you get down to 40 Knots or so, but you don't always expect to roll over when your angle of climb gets to be a little too much. But that's a limitation of WWI flying machines and not an arbitrary glitch in the program. I would liKe to have seen some sort of spirit level to let you Know when the plane is angled too much. Sometimes it is difficult to tell just by looKing out the cockpit, particularly in view of the fact that primary reference points are the sun, a distant horizon and layer of clouds and widely scattered objects on the ground.

Dogfights are the order of the day for pilots of the Spad XIII. and it doesn't let you down. Whether you are trying to shoot down the ten observation balloons (while avoiding heavy flaK), chasing a German two-seater or preparing to bomb a German airfield the challenge of Keeping your plane under control while taKing aim is engaging. With only 1000 rounds of ammunition in your twin VicKers machine guns marKsmanship counts for much. You don't want to shoot when the targets are too far away but if you wait until they're close they may get you first. Tiny bursts of white light represent machine gun fire from the enemy planes. Hitting balloons is easier because they're stationary but what does it matter if the Spad is hit by machine gun fire from a two-seater or flaK from the ground? The same goes for

bombing the airfield. It, too, is protected by flaK batteries. (Since there s no bomb sight, the only way to score a hit is to drop the bombs from a low altitude. In fact, you can't see them explode from an altitude greater than 1,000 feet.)

The reward for a successful engagement with the enemy is to watch as the shredded balloon or aircraft plummets to the ground. There is a score-Keeping feature that tallies the number of hits you've made but I found it to be almost irrelevant.

Although there are definite visual cues confirming hits you maKe, damage to the Spad shows up as loss of performance of the aircraft. If, all of a sudden, the Spad starts spinning wildly toward the ground and pulling backon the stick doesn't do any good, you can be pretty sure that you tooK a heavy hit. SmoKe erupting in the cocKpit would have been a nice touch to confirm that the Spad has been hit. Some visual confirmation of minor hits would at least allow you to head backfor the airfield to maKe repairs. I found that in most cases, by the time I realized the Spad had been hit, the damage was so serious that I couldn't return to the airfield. Those who pay more attention to the performance of the Spad than I have the option of returning to the airfield for repairs and rearming.

Ease of Use: To get the most out of Spad XIII you have to use the Keyboard rather than a joystick. Since 25 Keys are used to control everything from views to the rudder you have to be familiar with their locations before you can gain full control of the aircraft. Fortunately, the documentation is arranged so that a Keyboard template occupies the lower half of a page while a diagram of the cocKpit instrument panel occupies the top half.

Documentation: The manual is excellent. The 20-page booKlet includes plenty of diagrams and historical and performance data on the Spad XIII. It's written in an instructional style that could pass for the real thing.

Value: I thinK this is the best flight simulation program available for the TI. Bear in mind that it emulates a vintage aircraft and is therefore not directly comparable to the Dow-4 Gazelle or 4A-Flyer. While it is the equal of the Gazelle in terms of its aerodynamic realism, it is superior to it by virtue of its use of graphics. Both the Gazelle and Spad are superior to 4A-Flyer.

The only criticisms I have of it are minor, and mentioned above. It s not the ultimate flight simulation program in terms of what is possible on the TI, particularly in terms of the use of graphics. But the fact that it emulates a biplane rather than a modern, high-performance plane was a good decision by Not-Polyoptics. For what it tries to do, it succeeds quite well. And that's a lot more than I said about the last flight simulator I reviewed. YN

NEWSBYTES MICROPENDIUM Sep. 1987 Vol. 4 , No. 8

Not-Polyoptics has announced an upgrade to Spad III, its machine-Language flight simulator for the T199/4A, scheduled for release Sept. 15.

According to Michael V. Capobianco of Not-Polyoptics, Spad XIII MK. 2 is twice as fast as the previous version and more responsive.

New features have also been added, he says. In addition to the six "out of the cocK-pit" views, a removed viewpoint, showing the plane from an observer s standpoint, has been included.

"Better algorithms produce improved graphics and a more realistic flight performance," Capobianco says. "Keyboard/joystick interface has been augmented to allow new functions and greater flexibility. Finally, an optional 'Red Baron FoKKer' tri-plane opponent provides even more exciting dogfights."

Spad XIII, MK. 2 requires a T199/4A with Extended BASIC, 32K expansion memory and disK drive.

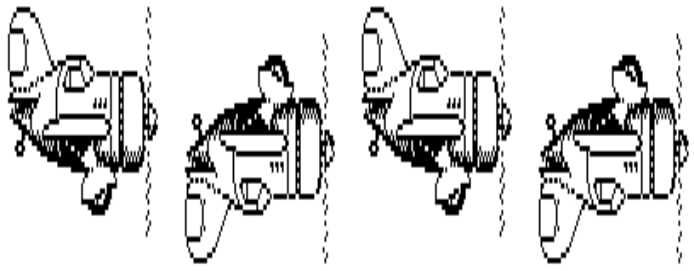
The program will retail for $29.95. according to the manufacturer. Registered purchasers of Spad XIII can obtain the upgrade by sending $6.95 plus $1 shipping and handling to Not-Polyoptics. YN

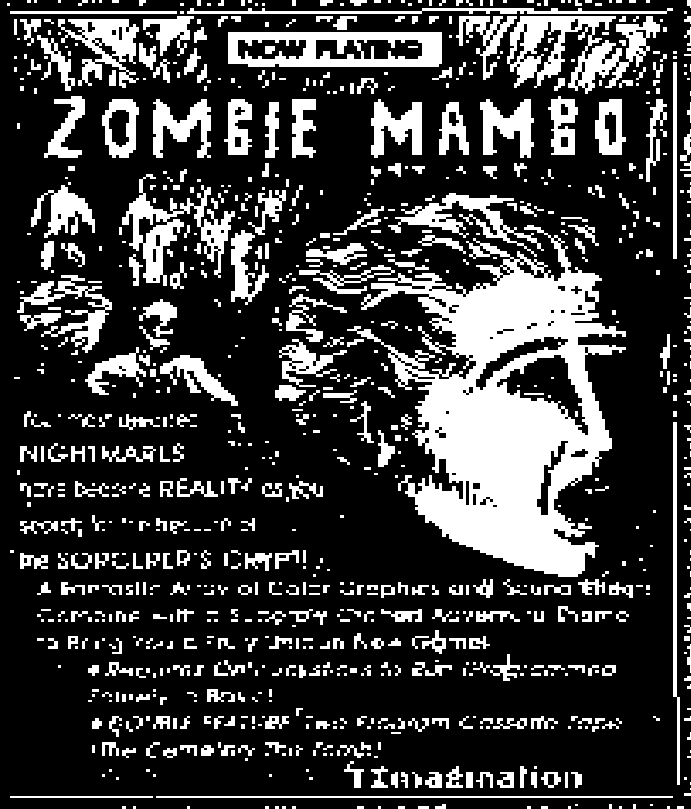
BHXP1 EXPERIMENTAL AIRCRAFT 1. 0

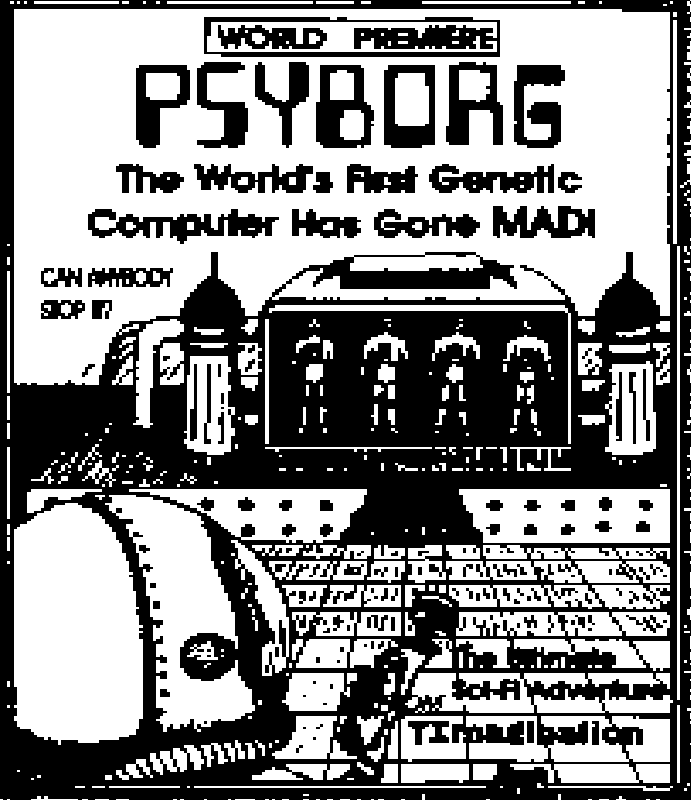
The BHXP1 is an experimental aircraft in the final testing stages before FAA approval. The airplane itself is a modified version of an airplane purchased in 1984. The plane has been restructured to handle higher speeds and has been fitted with a much more powerful engine. The instruments have been replaced with larger, easier to read models and DME (distance measuring equipment) has been added.

THIS PROGRAM IS BASED ON A 1982 TI BASIC PROGRAM FROM JOHN DOW.

ALL ASSEMBLY CODE AND MODIFICATIONS ARE BY BRUCE HELLSTROM.







TImagination, inc.
P.O. BOX 2606, FAIRFIELD, CA 94533

I am looking for a complete copy of the game PSYBORG, shown at left.

Anyone?

Thanks!
Sparkdrummer

NO PARKING

Yesterday's News Information

Yesterday's News is a labor of love offered as a source of pleasure & information for users of the TI-99/4A and Myarc 9640 computers.

TI-99/4A HARDWARE

TI99/4A COMPUTER
MODIFIED PEB
WHT SCSI AND SCSI2SD
MYARC DSQD FDC
MYARC 512K MEMORY
HORIZON 1.5 MEG HRD
TI RS232
CORCOMP TRIPLE TECH
1 360K 5.25 DRIVE
1 360K 3.50 DRIVE
1 720K 5.25 DRIVE
1 720K 3.50 DRIVE

TI-99/4A SOFTWARE

PAGEPRO 99
PAGEPRO COMPOSER
PAGEPRO FX
PAGEPRO HEADLINER
PAGEPRO GOFER
PAGEPRO FLIPPER
PAGEPRO ROTATION
PIXPRO
PICASSO PUBLISHER
BIG TYPE
TI ARTIST PLUS
GIF MANIA

PC HARDWARE

COMPAQ ARMADA 7800
COMPAQ ARMADASTATION
SAMSUNG SYNCMASTER

PC SOFTWARE

DEAD WINDOWS 98SE
FILECAP
PRN2PBNS
IRFANVIEW
ADOBE DISTILLER
ADOBE ACROBAT

Yesterday's News is composed entirely using a TI-99/4A computer system. It consists of 11 PagePro pages which are "printed" via RS232 to PC to be published as a PDF file.