YESTERDAY'S NEWS

30 Years Ago...

Historical Information taken from Bill Gaskills TIMELINE

April 1990:

MICROpendium publishes V7N3 consisting of 40 pages.

Texaments releases Artoons for TI-Artist Plus!

MicKey Schmitt releases GETTING THE MOST FROM YOUR CASSETTE SYSTEM.

Bruce Harrison releases his Word Processor v2.0.   YN

SUBMARINE COMMANDER

MICROpendium
November 1986, Volume 3, Number 10
by John Koloen

As Submarine Commander it is your responsibility to guide your submarine under an ocean to seeK out and destroy convoys of enemy ships. But don't thinK the enemy won't fight bacK, its convoys protected by destroyers just waiting to ram your submarine or send you to the bottom in a barrage of exploding depth charges.

This game was released for the TI about mid-year but has been available on other machines for a couple years. It combines colorful graphics and realistic sound into an entertaining strategy/action game that is filled with ups and downs, or submerging and surfacing. (Distributors include Tenex Computer Express and Arcade Hardware of Manchester, England.)

Performance: This game can taKe a long time to play - more than an hour is not unusual. Impatient types may find it frustrating. Despite this, a lengthy game is not necessarily indicative of a successful game. Scores are based on the amount of enemy tonnage sunK and whether you are able to sinK all the convoys at the given sKill level (there are three levels of play). Points are deducted depending on the number of torpedoes used (there is a finite number available), amount of fuel used (also finite), the amount of damage incurred (it can be repaired) and whether you are destroyed (which can be the result of running out of oxygen, being hit repeatedly by enemy depth charges or decK fire, having the hull crushed by descending to an excessive depth, etc.). There are actually a lot of ways to lose, and only one way to win - by sinKing every enemy convoy.

HAPPY BIRTHDAY

Yesterday's News
Has been published every
month since April 2016

TI CLASSROOM

TIPS FROM THE TIGERCUB NUMBER 11

By Jim Peterson

Last month's challenge was to unfurl the U.S.flag (with 49 stars), from the mast out in 2 lines of Extended Basic

```
100 CALL CLEAR :: CALL COLOR
(2,16,5,3,16,16,4,7,7):: A$(
1)="*******080808" :: A$(2)=
RPT$("80",7):: CALL CHAR(33,
RPT$("01",8)):: CALL VCHAR(4
,4,33,20)
110 FOR C=5 TO 22 :: X=1+ABS
(C>11):: FOR T=1 TO 13 :: CA
LL VCHAR(5+T,C,ASC(SEG$(A$(X
),T,1))):: NEXT T :: NEXT C
:: GOTO 110
```

One of the previous challenges was to write the Extended Basic statement IF X=1 THEN Y=7 ELSE IF X=2 THEN Y=33 ELSE IF X=3 THEN Y=19 ELSE IF X=4 THEN Y=21.My solutin was Y=VAL(SEG$("07 331921",X*2-1,2). Jim Johnston in the K-3 User's Group newsletter came up with a method which is better because it does not require that the values of X be in sequence:
Y=ABS((7*(X=1))+(33*(X=2))+(19*(X=3))+(21*(X=4)))

Proving once again that there is more than one way to skin the cat, and often a better way — although the cat might not agree.

Advice to disk-drivers — Keep and eye on those little tabs of silver tape that you use to cover the write-protect notch on your disks. They tend to become dog-eared from bumping against the slot of the drive. I recently heard a horror story about one of those tabs that came loose and got into the drive!

The following Menu-loader or Auto-booter was originally published by A. Kludge in the 99'er Vol. 1 #4. Marshall Gordon and Thomas Boisseau greatly improved it and published it in the Atlantic 99/4 UG newsletter Vol. 2 #1. I have no idea how it works, but have managed to modify it so that it will catalog up to 99 programs on a disk, stopping for input after each 19 are listed, or stopping whenever any key is pressed; I also added a delete option, requiring a repeated input to prevent error. It takes up only 8 sectors. If you have Extended Basic and Disk Drive, load this program under the file name LOAD. It will then automatically run whenever you select Extended Basic, will list all the programs on the disk, and will run whichever program you select.

(This program was successively updated in Tips #15, 18, 20, 22, 23, 24, 27, 28, 29, 30 and 32. - Ed.)

```
100 OPTION BASE 1 :: DIM PG$
(99),T$(5):: CALL CLEAR
110 T$(1)="DIS/FIX" :: T$(2)
="DIS/VAR" :: T$(3)="INT/FIX
" :: T$(4)="INT/VAR" :: T$(5
)="PROGRAM"
120 IMAGE ##
130 DISPLAY AT(1,9)ERASE ALL
:"DISKETTE MENU"
140 ! IF YOU HAVE MORE THAN
ONE DISK DRIVE, DELETE THE !
IN LINE 150
150 ! DISPLAY AT(12,6):"DISK
? (1-3):" :: ACCEPT AT(12,19
)SIZE(-1)VALIDATE("123"):D$
:: D$="DSK"&D$&"."
160 D$="DSK1." :: OPEN #1:D$
,INPUT ,RELATIVE,INTERNAL ::
 INPUT #1:N$,A,J,K :: DISPLA
Y AT(1,1)ERASE ALL:SEG$(D$,1
,4)&" - DISKNAME= "&N$:
170 DISPLAY AT(2,1):"AVAILAB
LE=";K;"USED=";J-K;"PROG  FI
LENAME    SIZE     TYPE":"___
_____" ::
I=0
180 FOR X=1 TO 80 :: IF X/20
<>INT(X/20)THEN 210
190 DISPLAY AT(24,1):"TYPE C
HOICE OR 99 FOR MORE" :: ACC
EPT AT(24,27)VALIDATE(DIGIT)
:K :: IF K=99 THEN 200 :: IF
 K>0 AND K<X+1 THEN 360 ELSE
 190
200 X=X+1 :: CALL VCHAR(1,2,
32,48)
210 I=I+1 :: IF I>127 THEN K
=X :: GOTO 300
220 INPUT #1:P$,A,J,B
230 IF LEN(P$)=0 THEN 270
240 DISPLAY AT(X+4,2):USING
120:X :: DISPLAY AT(X+4,6):P
$ :: PG$(X)=P$ :: DISPLAY AT
(X+4,18):USING 120:J :: DISP
LAY AT(X+4,22):T$(ABS(A))
250 CALL KEY(0,KK,ST):: IF S
T=0 THEN 260 :: FLAG=1 :: GO
TO 280
260 NEXT X
270 DISPLAY AT(X+4,1):" " ::
 DISPLAY AT(X+4,2):USING 120
:X :: DISPLAY AT(X+4,6):"TER
MINATE" :: DISPLAY AT(X+5,2)
:STR$(X+1)&" DELETE?"
280 DISPLAY AT(X+6,1):"    C
HOICE"
290 ACCEPT AT(X+6,16)SIZE(2)
VALIDATE(DIGIT):K :: IF K<>X
 AND K<>X+1 OR FLAG=1 THEN 3
50
300 IF K=X THEN CALL CLEAR :
: CLOSE #1 :: END
310 DISPLAY AT(X+5,11)SIZE(1
8):" #?" :: ACCEPT AT(X+5,15
)SIZE(2)VALIDATE(DIGIT):KD :
: IF KD<1 OR KD>X-1 THEN 310
320 DISPLAY AT(X+6,1)SIZE(28
)BEEP:"VERIFY - REPEAT DELET
E #" :: ACCEPT AT(X+6,27)SIZ
E(2)VALIDATE(DIGIT):KD2 :: I
F KD2<>KD THEN 340
330 DELETE "DSK1."&PG$(KD)
340 CLOSE #1 :: GOTO 130
350 IF K<1 OR K>99 OR LEN(PG
$(K))=0 THEN 270
360 CLOSE #1
370 CALL INIT :: CALL PEEK(-
31952,A,B):: CALL PEEK(A*256
+B-65534,A,B):: C=A*256+B-65
534 :: A$=D$&PG$(K):: CALL L
OAD(C,LEN(A$))
380 FOR I=1 TO LEN(A$):: CAL
L LOAD(C+1,ASC(SEG$(A$,1,1))
):: NEXT I :: CALL LOAD(C+1,
0)
390 RUN "DSKX.1234567890"
```

Come to think of it, if you have more than one disk drive you will also have to delete the first statement in line 160, and modify line 330.

Here's a memory-saver for you — put your data in strings instead of data statements. My "Hangman Plus" program was only 7764 bytes long but it contained a vocabulary of 315 words in data statements. After reading these into an array, it had too little working memory left, and paused too often for garbage collection After changing all the DATA statements to strings, it runs without stalling even though the number of words was increased and an array of 50 is still dimensioned for user input of words. When I loaded the original version in Extended Basic with the Memory Expansion and asked for SIZE after the DATA had been read in, I found that I had 14756 bytes of program and 7669 bytes of stack free. In the version with DATA in strings, at the same stage in the program I had 14874 bytes of program and 11310 bytes of stack

free P a saving of 3730 bytes! And another advantage is that there is no delay waiting for all those words to be read into the array. However, pulling DATA out of a string is undoubtedly a bit slower, so this method should not be used when speed is of primary importance.

In the "Hangman Plus" program, I used lower case letters as dividers between the upper case words. To pull words at random, I randomly selected a string and a position within the string, using the POS of the lower case letter to find the word. The following is a much abbreviated example:

```
100 M$(1)="aJOHNbJOEcCHARLIE
dMIKEeLARRYf"
110 M$(2)="aGEORGEbPETEcCHRI
SdDONeRALPHf"
120 X=INT(2*RND+1)
130 Y=INT(5*RND+97)
140 X$=SEG$)M$(X),POS(M$(X),
CHR$(Y),1)+1,POS(M$(X),CHR$(
Y+1),1)-POS(M$(X),CHR$(Y),1)
-1)
```

It is of course essential that all the strings contain the same number of elements of DATA. If lower case letters are needed, the separators can be ASCII codes 129 thru 154, obtained by holding down the CTRL key while typing the alphabet - It's a bit hard to keep track of those, because they're invisible! Numeric DATA can be also be stored, using the VAL function to convert it to numeric after it is pulled from the string.

You probably already know this, but you don't have type in the blank spaces before and after the :: in multiple statements in Extended Basic. Just run everything together 100 CALL::RANDOMIZE::FOR D=1 TO 100::NEXT

D and the computer will separate it for you, shoving statements into additional lines if necessary.

OUT OF MEMORY

HAPPY HACKIN'
Jim Peterson
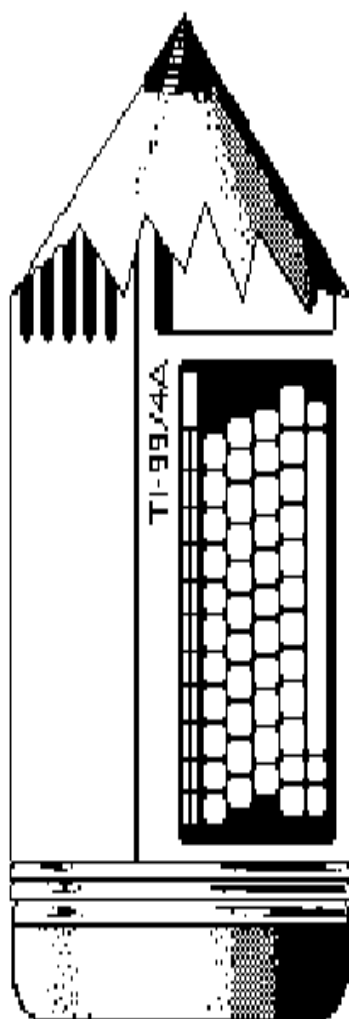
Submarine Commander uses four screens or views: the main control panel, a sonar view, a view of the surface through a periscope and a map depicting your position relative to land and convoys. The main panel includes numerous readouts reporting depth, position, fuel and battery status, speed, etc. Damage indicators are used to gauge the condition of the hull, engines indicators, controls and instrumentation. In the sonar view, the screen is filled with a sonar screen that depicts the position of enemy ships relative to the submarine. The periscope view shows the ships on the surface. It can be used only at depths of less than 40 feet. The map is used as an aid in stalking or avoiding convoys, providing a long-range view of the battleground. One may toggle between these various screens often and rapidly. Sound is used to good effect in simulating the sound of sonar, and depth charges. Not only do you hear the depth charges explode, but you hear them as they enter and descend through the water toward the submarine.

As commander of the submarine you control the speed of the vessel, its heading and depth. The longer you remain on the surface, the stronger your batteries become and the faster damage is repaired. While submerged you must keep a close eye on the amount of oxygen remaining, among other things.

Part of the reason I found the game to be engrossing is the wealth of details one has to keep aware of. It's not enough to track convoys, you have to chase them. Even then, they may seek refuge in a harbor and even trap you if you follow them in. Often, you will find yourself under fire from one vessel while attacking another, giving you just enough time to release one torpedo before crash-diving to avoid being sunk, If the enemy scores enough hits before you are able to submerge to the relative safety of the ocean's depths, you may find yourself in a waiting game, your engines and controls virtually unusable, your oxygen supply diminishing and the convoy hovering above you, visible as blips on the sonar screen.

Ease of Use: Rudder control, surfacing and diving may be controlled by a joystick or from the keyboard. All other functions are controlled by the number keys. The game comes with a keyboard template for reference. There is also a pause key for those times when you need a break from the tension of submarine warfare.

Documentation: The manual consists of two pages of compact print. It contains the essentials, and that is all. Much of what you need to know about playing the game, such as how to interpret the hydrophone chart, will come through trial and error.

Value: I enjoyed playing this game when I received it and continue to play it periodically. I am particularly enamored of the strategy that must be considered before

launching an attacK on an enemy convoy, the stalKing that has to be done and the fact that patience may be your best ally.

What maKes Submarine Commander a great deal for game players has nothing to do with the game itself. It's in the pacKaging. True, you get Submarine Commander; but included on the disK are Computer War and River Rescue, both of which are well-designed and entertaining games. YN
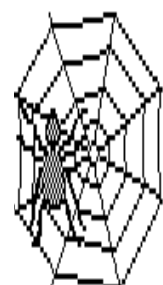
FUNNELWEB V4.40

Lima Ohio User Group
By Charles Good
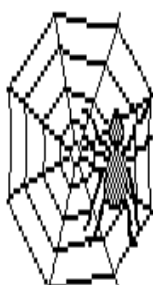
Although some releases of v4.40 have the date May 31/ 91, debuged v4.40 files were not actually available until early August. This version contains the new enhancements demonstrated by me at the May 18 Lima MUG Conference, plus a lot more! 80 column users will particularly benefit from some of these new features. As stated by Harry Brashear in the June 1991 issue of Micropendium, the existence of 80 column Funnelweb really justifies the purchase of an 80 column card for the 99/4A. In his letters to me, Tony McGovern suggests that this may NOT be the final Funnrelweb update. At some future date he hopes to completely rewrite the Funnelweb text/program editor.

Mainly because of new extensive doc files, v4.40 will no longer fit on one DSDD disK unless it is partially archived. The system files (both 40 and 80 column) occupy 709 disK sectors and the docs fill another 874 sectors. Thus our initial distribution with this newsletter is partially archived. Enough files are left unarchived to let you immediately try out the major features of v4.40, including the 40 column EDITOR and DISK REVIEW. Archiver is on disK and can be booted from Funnelweb to unpacK the rest of the files. UnpacKing can be done, and Funnelweb v4.40 can be used on a system with only SSSD drives. We are including supplementary files, not part of the official Funnelweb v4.4 pacKage, as space permits. These supplementary files include DSKU v4.2, DM1000, and foreign language character sets.

Upgrading from earlier Funnelweb versions is easy. You can use your old SYSCON file to quickly configure v4.40. Load your old SYSCON into v4.40's Configure program and then press BACK and "install" the configuration into v4.40's LOAD and FW files. You cannot safely use your old user lists (older versions of files UL and D1) directly with the new version. However, the v4.40 -READ-ME file gives easy and explicit directions for transferring data from your old UL and D1 files to the v4.40 UL and D1 user lists.

THE NEW FEATURES ADDED SINCE v4.31

SUPPORT FOR DSKU FILE COMMENTS: Many TI user groups use DSKU file comments to annotate their software libraries. Now these comments can be copied and (in 80 comumn DISK REVIEW) viewed and edited on screen. Normal file by file disK copying does not transfer DSKU file comments to the destination disK. Until now, the only way to copy these comments has been to use John Birdwell's DSKU program, or to use a whole disK file copier. Now Funnelweb will copy these comments when files are copied from within Funnelweb's 40 and 80 column DISK REVIEW. Tagging files and then asKing for an action (ctrl/A) from DISK REVIEW now has a new option N(otes). First you C(opy) all the tagged files to the destination disK by pressing "C". Then pressing "N" will transfer all DSKU file comments from the source disK to the destination disK. This is a two step process.

From 80 column DISK REVIEW you can also directly read these DSKU file comments on screen, and you can also edit them or create new comments where none existed. You can't read or create file comments directly from the 40 column DISK REVIEW, but you can use DISK REVIEW 40's sector editing capabilities to read/edit/create such file comments. Very specific instructions on how to do this are included in the 40 column DISK REVIEW documentation.

MULTIPLE USER LISTS SIMULTANEOUSLY ACCESSABLE AND DISPLAYED ON SCREEN: The supplementay user lists, each listing accessable from Funnelweb's central menus as USER LIST or DISK UTILS have been available in earlier releases of funnelweb. They are nice because the files called from these lists can have a file name of up to 10 characters and a path name of almost any any length, and the files can and can reside in multiple drives hard drives or ramdisKs. These user lists are great for booting software from hard disKs or from multiple drive systems that have "resident" disKs sitting all the time in specific drives (the poor man's hard drive). Each USER LIST can have up to 8 programs that RUN with the press of one Key.

There is nothing new in the above paragraph. What is new is the ability to display up to three (40 column systems) or six (80 column systems) user lists on screen sumultaneously when USER LIST is selected from Funnelweb's central menu. This display can also be obtained on power up if you CONFIGURE Funnelweb to immediately boot its USER LIST. You can then move the cursor with the arrow Keys over to the program you want to boot and press <enter> to run the program. Thats right folKs, a simultaneous display of up to 24 (in 40 columns) or 48 (in 80 columns) program names instantly bootable from any drive.

What you do is create separate user lists from within CONFIGURE using Funnelweb v4.40's UL file as a template and then save each user list to the Funnelweb boot or TIW drive (as designated from within CONFIGURE) with a file name other than UL (such as UM, UN, etc). Then taKe Funnelweb v4.40's ML (or ML80) file, rename it UL, and place it on the Funnelweb boot drive. When you select USER LIST from the TIW Funnelweb central menu, ML or ML80 reads all the user lists and displays all their file names on screen!terson

ENHANCED 80 COLUMN SHOW DIRECTORY: 80 column users already Know about the ability to store multiple large text files in memory for rapid viewing with 80 column DISK REVIEW. Now you can do almost the same thing from within the 80 column text/program editor. From within Show Directory you can V(iew) a second text file while the text file you are editing remains in memory. The V(iew)ed file is displayed as one 80 column page of text at a time. This is not new to v4.40.

What is new is that up to 24 screens of text from one or from several V(iew)ed files CAN BE STORED IN MEMORY for almost instant access without further disK activity. I have a 44 sector TI Writer help file that taKes up only 8 screens. I can store other text files in the remaining 16 screens of the Show Directory V(iew) text storage buffer. Once I load my help file into memory, I can rapidly switch bacK and forth between the text editor and show directory display buffers for viewing purposes. From the edit buffer I can press SD, V(iew) my help file without bothering to boot a disK directory, go bacK to the text I am editing, and then later instantly bring up my help file again as needed.

No, you can't rapidly exchange text between the 80 column edit and SD buffers without first saving your edit buffer text to disK. And no, this extra text V(iew) storage buffer is not available to 40 column users. Some of the extra VDP memory associated with 80 column cards is used to store the V(iew)ed text, and this memory does not exist on 99/4A systems without an 80 column card. 40 column users can still V(iew) text from Show Directory one screen at a time, but there is no memory buffer for the V(iew)ed text.

COMBINATION 40/80 COLUMN EDITOR: There are times when 80 column users would benefit from a 40 column editor, for example when preparing documents designed to be displayed on a 40 column screen. The new 80 column text/program editor can be switched bacK and forth between a 40 and 80 column display. Of course you need an 80 column card to get an 80 column display. 40 column only users still have a separate 40 column only editor. The 40/80 column editor is combined with an 80 column only Show Directory screen with all the new V(iew) enhancements described above.

IMPROVED ERROR HANDLING: Funnelweb v4.40 should now be compatible with grom library devices such as O.P.A.'s gizmo, the Mechatronic gram card, and an enhanced gramulator.

When loading DF80 software, Funnelweb will now display the names of any unresolved REFs or duplicate DEFs that are encountered.

ACCELERATING CURSOR: The flashing cursor autorepeats and also accelerates as a single Key is held down. This acceleration is new.

ASSEMBLY "PROGRAM" FILE MAKE FROM SCRIPT LOAD: A greatly enhanced Script Loader (file SL), when called from LOADERS option of Funnelweb's central menu, can assemble a linKed group of DF80 object code files into runable assembly PROGRAM FILES. Extensive documentation describing the use of SL tells how. Tony Mcgovern says this feature has been used extensively by him in the creation of Funnelweb v4.40. The new Script Load should be useful to those creating very large assembly programs from a series of separate DF80 object files as is often done in the development of c99 software. If you have any long groups of DF80 files that taKe forever to load (such as early versions of the games TENNIS and ARCTURUS) you might try running them through Script Load to convert them to quicK loading EA PROGRAM files.

ASSEMBLY LANGUAGE PROGRAM SERVICES: Funnelweb loads some special assembly language callable routines (with EQU >xxxx). These routines can be used by programmers who create source code that is designed to run from the Funnelweb environment. Many of these routines have been available in earlier versions of Funnelweb. They are now fully documented and available to the programming "public". Some of these routines include:

DSRLNK, that is compatible with multiple RS232 cards.
KSCANA, an enhanced KSCAN.
DELSPR, shuts off the sprite list for quicK return to text mode.
VMBWD, a VMBW that saves space by ignoring nuls in a fixed length data value.
VMBRD, the VPD read version of VMBWD.
VFILL, fills a blocK of VDP RAM with a single byte value.
VSTRW, writes a string to VDP.
CURSOR, an enhanced cursor routine.
DSRREN, a direct DSR reentry from saved values.
SETGRD, sets GROM address so that module library banKing is supported.
CFILE#, sets the number of open files, as in CALL FILES
RDDEV, builds a PAB in VDP.   YN

99'er
November
1982
Vol. 2
No. 1

by
W.K.
Balthrop

DEFEND THE CITIES

I found myself perched high atop one of New York's tallest skyscrapers. There I was nervously waiting for the expected alien attack to begin. My job was to command one of the new building-launched interceptor ships. Our mission: to detonate falling bombs and to eliminate at whatever cost, the attacking ships.

Since there weren't many of us skilled "city pilots" around, I was supposed to be teleported to Los Angeles to help the West Coast defense team if I survived the five fierce attacking ships attempting to make sauce out of the Big Apple . . .

Well, I did somehow manage to save NY and LA, and expected a big ticker-tape parade . . but those down-right nasty aliens knew other-wise: Three more cities had to be successfully defended before I could hang up my uniform and rejoice in the knowledge that Earth was safe.

Two versions of Defend The Cities are available. The first version is written in TI Extended BASIC. The second version is written in 9900 Assembly Language to be used with either the Mini-Memory cartridge, or Expansion Memory peripheral box or card in conjunction with the Editor/Assembler cartridge.

If you have Extended BASIC, you'll find this one of the better games being offered in the arcade catagory. The game makes good use of sprites and graphics to keep you interested. Defend The Cities can be played with the keyboard or joystick. The joystick is suggested, however, because the action is a little difficult to control on the keyboard.

Intersoft has somehow gotten around one of the biggest problems in the use of sprites with Extended BASIC. Normally, sprite coincidence is very difficult to check because of the slowness, but this game uses fairly fast sprites and makes very few mistakes when checking for hits.

One problem I ran into while playing the game was the slow response to keyboard input when moving or firing. I often found myself colliding with an alien while waiting for the keyboard to be scanned with the result that the game would end prematurely.

The Mini-Memory assembler version of Defend The Cities can be loaded from its cassette tape and stored in the Mini-Memory cartridge for instant use. With this more sophisticated implementation, the original Extended BASIC version has been transformed into the kind of game you might pump quarters into all day at a commercial arcade. The game scenario is identical, but the action is much faster. The key response is almost instantaneous, and errors are non-existant in checking for hits, or collisions. If this is a sample of programs to come that can fit into and be run from the Mini-Memory cartridge, you will find this TI Command Cartridge (suggested retail, $99.95) a great investment.

I did encounter a couple of inconveniences with both versions. One was that the player's ship can wrap around the screen. Due to inexperience in ship handling, the ship will occasionally wrap around the bottom or top of the screen, placing the ship off screen for a period. You can't fire from there and it sometimes takes awhile to find your way back into action.

The second inconvenience encountered was when a player's ship is left drifting out of control each time the alien releases a bomb. After the bomb is dropped, control returns and the play continues. The problem arises when your ship drifts into a bomb or the alien ship just as play resumes making the game come to an unfair finish. I felt like the aliens had cheated.

One really super feature in the Mini-Memory version of Defend The Cities is that the high score is automatically saved in the module. If you ever want to prove to your brother-in-law or friend down the street that you really did score two trillion points, just take your Mini-Memory Cartridge to his house, plug it into his TI Home Computer, and there on his screen will be your intimidating score!

The documentation for Defend The Cities is contained in a nine page pamphlet. The start up procedures and rules are well written and easy to understand. Overall, this was a rather enjoyable arcade game which should have a large appeal to players of all ages.   YN

THEON RAIDERS

MICROpendium
July 1984
Volume 1, Number 8

by John Koloen

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

Theon Raiders went on the market in 1983. Except through mail-order it has not been readily available. And that's too bad, because I found this game to be more absorbing than any of the TI cartridge-based space games, including

Parsec. If I had to choose between having Theon Raiders or Star Trek by Sega Enterprises Inc. in my software collection, I'd have a tough time making the decision.

This game comes in versions for the Editor/Assembler cartridge and the Mini-Memory cartridgge. This review is based entirely on the Editor/Assembler version.

Performance: Theon Raiders is designed for joystick jockeys. This is a shoot-'em-up, no two ways about it. Sure, you've got to navigate your vessel, decide on the velocity and every once in a while go into warp drive and hyperspace, but when all is said and done your job is to shoot down the menacing hordes of aliens who are intent on destroying your space station.

After loading the game a brief title sequence is displayed. The screen consists of a view of space from the bridge of your star ship and a small blocked off area at the top left of the screen that serves as your radar. In the center of the radar scanner is a circular space station. Your ship is inside the space station when the action gets under way. Surrounding you are several enemy cruisers.

On the right top portion of the screen are four numeric readouts. The first one reports your compass bearing, the second your power supply and the third your velocity. The fourth one works only in the Editor/Assembler version and only at difficulty level three. It reports on the amount of damage your photon torpedoes have suffered from direct hits. There are seven torpedo banks and each direct hit knocks out one of them. They can be restored by reducing velocity.

After choosing the difficulty level, 1-3, the game gets under way. (Difficulty level one should be accessible to most players. Level three is very tough.)

To leave the space station you must accelerate the star ship. Pressing the number 9 key increases velocity while pressing the 0 key reduces it. Reaching a velocity beyond 100 automatically puts you into warp drive. At warp speeds you can outrun the Theon battle cruisers as well as their torpedoes. Direction is controlled by using the joystick. Moving it to the left causes the compass bearing to decrease while moving it to the right causes it to increase.

After leaving the space station, you must attack the enemy cruisers before they are able to reach the station. You set your compass bearing and then adjust your speed. As you approach a Theon cruiser you see orange blips approaching you. These are its torpedoes and the nearer they get the larger they become. You must destroy them with your torpedoes to avoid being hit by them, or you can dodge them. Eventually you may reach the cruiser itself, a white blip. You destroy it in the same way as you destroy the Theon torpedoes. The Theon cruisers will crash into your ship if you fail to destroy them, thus ending the game.

Generally, you fire your torpedoes in bursts, and direct them into the targets using the joysticks. Once you've locked onto a Theon the action is furious. Your wrist will get cramps trying to fend off the oncoming torpedoes and Theons.

After destroying a Theon you must change your bearings again and go after the next one until you've rid the sector of all Theons. Then you go into hyperspace and emerge into another sector, complete with a space station and more Theons. The primary difference between the sectors is that the color of your torpedoes changes.

Every hit you score on a Theon or its torpedoes will add power units while every torpedo you fire depletes your power supply. The amount of power that is restored and depleted is also determined by the velocity of your star ship.

When you've finally run out of power units the game ends, with your score appearing at the bottom of the screen. The high score appears at the top of the screen. You may replay or exit the game at this point. I think you'll choose to replay.

Theon Raiders creates an excellent three-dimensional effect utilizing photon torpedoes, approaching enemy torpedoes and a background of stars. The game play is flawless. Written in assembly language, all input, whether through the joystick or keyboard, bears instantaneous results. Sound effects are realistic and well done. There are sounds for the firing of torpedoes, explosions and a whirring-type sound when you rotate the position of the star ship.

Ease of Use: Aside from following the loading instructions, and learning how to control your velocity and direction, this game is easy to use, though difficult to win. Even at the easiest difficulty level, at high speeds, everything happens very quickly. The challenge is not in figuring out how the game works but in figuring out how to defeat the Theons, which is as it should be.

Documentation: Theon Raiders comes with a 10-page pamphlet that includes loading instructions for all versions. It provides an adequate description of how to play the game.

Value: Because this game is written in assembly language it is directly comparable to any other space game available for the TI. Although I am not familiar with all such games, I know of none that is better designed or executed. And, it is priced right at $24.95. Seeing games of this quality developed by third-party programmers is indeed encouraging.　　　　　YN

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.

NOW PLAYING