```
**************************************************************************
*                                                                        *
* DDD   III   GGG   III   SSS   Y   Y  N   N  TTTTT                       *
* D D   I    G      I    S       Y Y   N   N    T                  1   000 *
* D D   I    G      I    S        Y    NN  N    T                 11   0 0 *
* D D   I    G GGG  I     SS      Y    N N N    T         v   v   1 1  0 0 *
* D D   I    G G    I      S      Y    N  NN    T          v v     1   0 0 *
* DDD   III   GGG   III SSSS      Y    N   N    T           v  .   1 . 000 *
*                                                                        *
**************************************************************************
```

THIS PROGRAM IS FAIRWARE.
IF YOU LIKE IT, AND IF YOU LIKE TO
SEE IN THE FUTURE FURTHER
DEVELOPMENT OF DIGISYNT,
PLEASE SEND WHAT IT IS WORTH
TO YOU TO:

       STEFANO BONOMI
       VIA SACCHI 21
       37124 VERONA
       ITALY

YOU ARE ENCOURAGED TO DISTRIBUTE DIGISYNT FREELY PROVIDED YOU CHARGE NO MORE
THAN MEDIA AND DISTRIBUTION COSTS. ALL COPIES OF DIGISYNT DISKETTE MUST INCLUDE
THIS DISCLAIMER.

IF YOU DEVELOP APPLICATIONS USING DIGISYNT YOU MAY MARKET THEM PROVIDED THAT
THE SOFTWARE AND DOCUMENTATION ACKNOWLEDGE THE USE OF DIGISYNT.

THIS SOFTWARE CARRIES NO WARRANTY, EITHER WRITTEN OR IMPLIED, REGARDING ITS
PERFORMANCE OR SUITABILITY. NEITHER THE AUTHOR NOR ANY SUBSEQUENT DISTRIBUTOR
ACCEPTS ANY RESPONSIBILITY FOR LOSSES WHICH MIGHT DERIVE FROM ITS USE.
HOW WORK DIGISYNT.

Digisynt was completely developed using TI EDITOR ASSEMBLER package on a 1983
TI 99 4A console connected with a
TI PERIPHERAL EXPANSION SYSTEM containing TI 32Kb expansion memory and two
DS/DD drives.
Digisynt it' s a program that allow you to digitize and synthesize every kind of
sound. It make use of the TI 99/4A standard cassette TI 99 input port in order
to read the sound you play with any connected sound players device.
Then it make an analogical to digital conversion so that you can
then play the sound or save it on disk. At this point it's also possible to
reproduce what you saved using TI EXTENDED BASIC or TI BASIC (if you have the
editor assembler cartridge inserted in your console).

NEEDED EQUIPMENT.

For use DIGISYNTH you need a TI 99/4A console with 32KB memory expansion, a
disk drive and EXTENDED BASIC or EDITOR ASSEMBLER cartridge. For digitize
sound you need also a cassette recorder and the standard cable
that you received with your console in order to connect the EAR socket to the
TI 99. This connection should be done simply like when you want to load a
program using a cassette player.

FILES IN THIS PACKAGE.

In this disk you will find following files:

- DS - this is digisynt main EA program file
- BASYNT - this is a subprogram for heard the digitized sound using TI BASIC
           with EDITOR ASSEMBLER cartridge
- EXSINT - like BASYNT, but it works with TI EXTENDED BASIC
- SUBMARINE - an example game that works with TI EXTENDED BASIC and shows you
             how use subprograms for play digitized sound
- README - this file
- BLAST, CAT, COW, DISCO, DOG, FROG, GAOV, HORSE, KAPOW, NAME, RECORD - examples
  of digitized sounds (with a little experience you will do better!)

HOW TO LOAD DIGISYNT.

You can load digisynt from EDITOR ASSEMBLER cartridge OPTION 5 with the program
name DS as well as from EXTENDED BASIC. Selecting EXTENDED BASIC from TI MAIN
SCREEN will load the program for you from drive 1.

MAIN MENU.

When loaded as above, the program will display on the screen the following menu:

1) DIGITIZE
2) SYNTETIZE
3) MODIFY DELAY
4) MODIFY MEMORY AVAILABLE
5) SAVE DATA ON DISK
6) LOAD DATA FROM DISK

1) DIGITIZE.

If your cassette recorder is properly connected, then selecting first option
from menu followed by ENTER, you will digitize what your cassette recorder is
playing. The sound will be recorded in defined available memory.
During computer operation, cursor will disappear from the screen; you must leave
your cassette player running since the cursor appear back on the screen.
At this point you can stop the tape.
NOTE:
Normally for a good reproduction quality, cassette recorder volume should not be
too high.

2) SYNTETIZE.

For listen what you have previously ditized or loaded from disk, You have to
choose option 2 from menu and then the program will reproduce the sound. The
sound will be played with the last modification of delay and memory space.

3) MODIFY DELAY.

The delay is a parameter that has a big effect on digitize-synthesize operation
it modify the program scan speed of the sound.
Increasing the time delay, will decrease the quality of sound, but
will increase the length of what you can digitize.
The default for delay value is 1, but you can change this
value as you want, from 1 to 9 (expressed in decimal form)
The delay value may also be changed after the digitalization, causing in
this case a modification of reproduction speed.

4) MODIFY MEMORY AVAILABLE.

DS allow you to use for your own data storage two areas; the first is located in
the lower memory expansion between 3000 and 4000, the largest is in the higher
memory expansion, between A000 and FFFF. Both of start and end addresses must be
given in hexadecimal notation. About memory, this program have two important
limitation;

- if you store sound in the low memory, you can't then reproduce it from BASIC
  or EXTENDED BASIC using SYNT subprogram.
- if you like to reproduce your sound in BASIC using SYNT subprogram, the start
  address of sound buffer must be at least >A500. Check also the data files on
  this disk with option 4 of digisynt, before you try it using SYNT subprogram
  from BASIC. Using SYNT subprogram from EXTENDED BASIC you don't have this
  problem.

Default memory is between >B000 and >E000.

5) SAVE DATA ON DISK.

If you choose this option from main menu, the program will ask you for the name
of the file in the form "devicename.filename", and then will save on disk your
stored data, as well the related values for memory used and delay factor.

6) LOAD DATA FROM DISK.

In this case the program will ask you for the name of the file and then it
will load digitized data.

NOTES:
- you can go back in every moment to MAIN MENU pressing BACK (FTCN+9)
- you can exit the program pressing QUIT (FTCN+=)

HOW TO PLAY DIGITIZED SOUND FILES IN TI BASIC AND TI EXTENDED BASIC USING BASYNT
AND EXSYNT.

These are the names of two files that allow you to synthesize the sound
you've digitized and saved using DS, from your own program in TI BASIC or
TI EXTENDED BASIC.
BASYNT is to use in TI BASIC environment and EXSYNT in
TI EXTENDED BASIC.
Once loaded, these files will give you two assembler subprograms for load and
play your sound files.
For use BASYNT, you must have the TI EDITOR ASSEMBLER cartridge inserted in your
console and you have to select TI BASIC from MAIN SCREEN.
Then you should type following instruction:

```
10 CALL INIT
15 REM *** this example
16 REM *** assumed that
17 REM *** DIGISYNT diskette
18 REM *** is in drive 1
20 CALL LOAD("DSK1.BASYNT")
21 REM * you will find the file
       * BSCSUP on TI EDITOR
       * ASSEMBLER disk labeled
       * part A
22 REM * it contains the standard
23 REM * E/A support routines.
25 REM * You can do a copy of this
26 REM * file also on your
27 REM * DIGISYNT disk
30 CALL LOAD("DSK1.BSCSUP")
```

And from TI EXTENDED BASIC:

```
10 CALL INIT::CALL LOAD("DSK1.EXSYNT")
```

Usage of subroutines is the same in TI BASIC as in TI EXTENDED BASIC.
The first routine allow you to read a digitized file previously saved from
DIGISYNT, and has the form:

```
CALL LINK("CHRG","DEVICENAME.FILENAME")
```

this will load the digitized sound file in memory at the memory address defined
in digisynt.
At this point you can play your sound with the statement:

```
CALL LINK("SYNT",VOLUME)
```

Where volume is a numeric variable expressed in decimal notation, and can be
between 15 (silence) and 0 (max volume).
Here follows a complete example of usage for these two routines:

```
10 CALL INIT
20 CALL LOAD("DSK1.BASYNT")
30 REM ** 20 CALL LOAD("DSK1.EXSYNT") USING EXB. **
40 CALL LOAD("DSK1.BSCSUP")
50 REM ** YOU DON'T NEED TO LOAD
60 REM ** BSCSUP IF YOU ARE
       ** WORKING IN
       ** EXTENDED BASIC
70 CALL LINK("CHRG","DSK1.KAPOW")
80 FOR VOLUME=10 TO 0 STEP -2
90 CALL LINK("SYNT",VOLUME)
100 NEXT VOLUME
```

```
*******************************************************************************
* P.S.                                                                        *
* While subprogram "SYNT" is working in TI BASIC or EXTENDED BASIC,           *
* interrupts will be disabled, and so automatic SPRITE MOTION will not take   *
* place during reproduction of digitized sounds.                             *
*                                                                             *
*******************************************************************************
```

On this disk you can also find an EXTENDED BASIC game that make use of these
subroutine. For load it, you should write the following statement:

```
OLD "DSK1.SUBMARINE"
RUN
```

You can also LIST this little game, so that you can see
further examples about the usage of DIGISYNT subroutines.

At the end, I hope that you like this program. I have also ideas for a new and
better version of it. Your opinion and suggestion about DIGISYNT will be
appreciated.