Trisyd Video Games Dynomite Sound Digitizer technical docs

Date Aug 8, 1990

This document file is explicitely for the use by purchasers of my Dynomite Sound Digitizer product. Use of this material by other persons who have not already ordered and received their Trisyd Video Games Dynomite Sound Digitizer product is strictly prohibited. This file is NOT public domain, I am releasing it in order to propagate the use of your sampled waveforms. It is not intended to allow direct competition with Trisyd Video Games. Failure to comply with my restrictions will discourage the release of detailed documentation of this kind in the future and ultimately dwindle our computers future.

Do not distribute this information file or any modified versions without first consulting with me. Please respect my wishes and I'll respect your rights to full use of my products.

Syd Carter
Trisyd Video Games

The next few pages contains the complete source code listing for the basplyr machine language overlay file. It was written for assembly using Z80MR, a CP/M (say TDOS??) type utility.

It details the three components required to reproduce the digitized waveforms on Adam. These are.. Voice channel init, Output and finally Sound off. Since Z80MR requires that your program starts at location 100h, I included an offset to place the machine language file where you would want it. This can be modified to place the basplyr program outside of memory which may already be used by other machine language programs.

```
;* Sound player program Basic overlay
;* Written July 31/90 version 1.0
;* By Syd Carter / Trisyd Video Games
;* For use with sound data files created using Trisyd Video Games
;* Dynomite Sound Digitizer.
;
;   Trisyd Video Games
;   26 Florence Cres
;   Toronto, Ontario.
;   Canada  M6N 4E4      (416) 769-6446
;

        ORG     0100h    ;Z80MR origin but actual address uses offset
;
OFFSET  EQU     27936-100H-5    ;Program residing in memory at 27936
SNDPRT  EQU     0FFH     ;SOUND PORT
OFF     EQU     0FFH     ;SOUND OFF DATA
;
BLOAD:
;basic front end identifier
        DW      START+OFFSET
START:
        JR      SNDTBM   ;POSITION TABLE
;
```

```
CLAIM   DB      'TVG90'  ;identifier (leave intact)!!
SNDLOC  DS      2        ;reserve 2 bytes for sound location poke
SNDLEN  DS      2        ;reserve 2 bytes for sound length poke
OPRAT   DS      1        ;reserve 1 byte for output rate
;
;
SNDTBM:
        PUSH    AF
        PUSH    BC       ;SAVE YOUR REGISTERS!
        PUSH    DE
        PUSH    HL
;moves sound table to a location in memory that starts exactly at a
;page boundry.  This allows for faster access to data
        LD      DE,SNDTBO+offset ;STORAGE LOCATION PADDED WITH 256 BYTES
        LD      HL,SNDTBL+offset
        LD      BC,0100H ;DON'T FORGET FOR VALUES OF FFH
        LD      E,0      ;EVEN PAGE
        LD      (SNDTBV+offset),DE;REFERENCE FOR LATER USE
        LDIR
;sound table in place, now install output rate
        LD      A,(OPRAT+offset)
        LD      (SPKRAT+offset),A
;now set up start and end vectors
;
PLBSND:
;Play sound block parms

        CALL    INITFQ+offset   ;FREQ INIT
        CALL    NOSND+offset    ;TURN OFF ANY GARBAGE
        LD      HL,(SNDTBV+offset) ;where sndtbl gets loaded
        LD      DE,(SNDLOC+offset) ;SNAG location
        LD      BC,(SNDLEN+offset) ;snag length
;
PLYSPK:
;Parameters set, play the tune.
        LD      A,(DE)   ;STORED SOUND
        CALL    OPSND+offset    ;SEND IT OUT
        INC     DE
        DEC     BC
        LD      A,B      ;Sound length inspection
        OR      C
        JR      NZ,PLYSPK       ;loop till done.
;
        CALL    NOSND+offset    ;enough noise already
;return to basic now.....
        POP     HL
        POP     DE       ;RESTORE REGISTERS
        POP     BC
        POP     AF
        RET     ;back to basic language now.
NOSND:
        LD      HL,(SNDTBV+offset) ;FOR SOME REASON IT NEEDS TO BE
RESTORED
        LD      A,OFF    ;turn off all three voices
        OUT     (SNDPRT),A      ;Turn off noise too
        CALL    OPSND+offset
        RET
INITFQ:
        LD      E,0
        LD      BC,0381h
```

```
INILOO:
        LD      A,C
        OUT     (SNDPRT),A
        ADD     A,20h
        LD      C,A
        LD      A,E
        OUT     (SNDPRT),A
        DJNZ    INILOO
ININOI:
        LD      A,11100101B     ;NOISE TYPE WHITE
        OUT     (SNDPRT),A
        RET
;
OPSND:
;outputs sound byte per table values
        LD      L,A       ;OFFSET INTO TABLE
        LD      A,(HL)    ;1ST VOLUME LEVEL
        OUT     (SNDPRT),A      ;1ST O/P
;remaining channels are encoded into table pointed to by HL
        INC     L
        LD      A,(HL)
        OUT     (SNDPRT),A
        INC     L         ;NEXT TABLE ENTRY
        LD      A,(HL)
        OUT     (SNDPRT),A
;
        PUSH    BC
SPKINS  DATA    06      ;ld b

SPKRAT:
        ds      1         ;output rate placement
OPLOO:
        DJNZ    OPLOO
        POP     BC
        RET
;
SNDTBV  DW      0000
;
SNDPAD  DS      100H      ;MUST USE FOR PROPER PADDING
;
SNDTBO  DS      100h      ;must reserve max of 100h here.
;
SNDTBL:
;Use this table to arrive at appropriate data for sound voices
;therefore use 9x bx AND dx

S00     DB      0BFH,0DFH,09FH,0BFH,0DFH
        DB      09FH,0BFH,0DFH,09FH,0BFH
        DB      0DFH,09FH,0BFH,0DFH,0DFH  ; cannot synch data at ends
        DB      09FH
S0      DB      0B0H,0D0H,090H,0B0H,0D0H
S1      DB      090H,0B0H,0D1H,090H,0B0H
S2      DB      0D1H,090H,0B1H,0D1H,090H
S3      DB      0B1H,0D1H,091H,0B1H,0D1H
S4      DB      091H,0B1H,0D2H,091H,0B1H
S5      DB      0D2H,091H,0B2H,0D2H,091H
S6      DB      0B2H,0D2H,092H,0B2H,0D2H
S7      DB      092H,0B2H,0D3H,092H,0B2H
S8      DB      0D3H,092H,0B3H,0D3H,092H
S9      DB      0B3H,0D3H,093H,0B3H,0D3H
```

```
SA      DB      093H,0B3H,0D4H,093H,0B3H
SB      DB      0D4H,093H,0B4H,0D4H,093H
SC      DB      0B4H,0D4H,094H,0B4H,0D4H
SD      DB      094H,0B4H,0D5H,094H,0B4H
SE      DB      0D5H,094H,0B5H,0D5H,094H
SF      DB      0B5H,0D5H,095H,0B5H,0D5H
S10     DB      095H,0B5H,0D5H,096H,0B5H
S11     DB      0D5H,096H,0B5H,0D6H,096H
S12     DB      0B5H,0D6H,096H,0B6H,0D6H
S13     DB      096H,0B6H,0D6H,097H,0B6H
S14     DB      0D6H,097H,0B6H,0D7H,097H
S15     DB      0B6H,0D7H,097H,0B6H,0D7H
S16     DB      096H,0B7H,0D8H,096H,0B7H
S17     DB      0D7H,096H,0B9H,0D7H,096H
S18     DB      0B9H,0D7H,097H,0B9H,0D7H
S19     DB      097H,0B9H,0D8H,097H,0B9H
S1A     DB      0D8H,097H,0BAH,0D8H,097H
S1B     DB      0BAH,0D8H,098H,0BAH,0D8H
S1C     DB      097H,0BAH,0DAH,097H,0BAH
S1D     DB      0DAH,097H,0BBH,0DAH,097H
S1E     DB      0BBH,0DAH,098H,0BBH,0DAH
S1F     DB      098H,0BBH,0DBH,098H,0BBH
S20     DB      0DBH,098H,0BCH,0DBH,099H
S21     DB      0BBH,0DBH,09AH,0BBH,0DBH
S22     DB      09AH,0BBH,0DCH,09AH,0BBH
S23     DB      0DCH,09AH,0BCH,0DCH,09AH
S24     DB      0BCH,0DCH,09BH,0BCH,0DCH
S25     DB      09BH,0BCH,0DDH,09BH,0BCH
S26     DB      0DDH,09BH,0BDH,0DDH,09BH
S27     DB      0BDH,0DDH,09CH,0BDH,0DDH
S28     DB      09CH,0BDH,0DDH,09DH,0BDH
S29     DB      0DDH,09DH,0BDH,0DEH,09DH
S2A     DB      0BDH,0DEH,09DH,0BEH,0DEH
S2B     DB      09DH,0BEH,0DEH,09EH,0BEH
S2C     DB      0DEH,09EH,0BEH,0DFH,09EH
S2D     DB      0BEH,0DFH,09EH,0BEH,0DFH
S2E     DB      09EH,0BFH,0DFH,09EH,0BFH
S2F     DB      0DFH,09EH,0BFH,0DFH,09FH ;end of installed data

        END

;That's it, if re-assembling, remove this text line and anything after it.

        ---END---
```