

3 W.F.MODE X1 X2 3 4 5 X6 7 X8 9 A B C D E F  
 INIT: .WORD 1000,1400,0,0,0,3400,0,5400,0,0,0,0,0,0,0  
 X1=RAMP SCALER; X2,X6,X8=WAVEFORM TABLE CENTRE ADDR.

MESS: .ASCII @ USE ARROWS OR TAB TO POSITION CURSOR, UP/DOWN TO CHANGE VALUES<15><12>  
 .ASCII / USE <> TO CHANGE INC VALUE; HIT "RETURN" TO PRINT; CNTRL C TO EXIT/<15><12>  
 .ASCII / HIT SPACEBAR FOR NEW ATTACK (AT RATE=INC) OR TO RESET IF CONTROL IS LOST/<  
 .ASCII / FOR THE FIRST 7 VARIABLES, THE FOLLOWING OPTIONS APPLY!/<15><12>  
 .ASCII / NEW VALUES MAY BE TYPED IN; THEY ARE PUT INTO EFFECT WITH "ENTER"/<15><12>  
 .ASCII / TO STORE A VARIABLE, USE "S"; TO REPLACE CURRENT WITH STORED VALUE, USE "R"/<1  
 .ASCII / TO SYNC VARIABLES USE + FOR POS.CORRELATION, - FOR INVERSE, "DELETE" TO OMIT/<1  
 .ASCII / NOTE: CHANGE DUR.RNG TO DEL.RNG WITH \ ; '=HARMONIC FREQ./<15><12>  
 .ASCII / SYNC VARIABLES CAN BE RAMPED (RAMP=NO.MILLISEC.BEFORE ADDING INC VALUE)/<15><1  
 .ASCII / USE "A" TO START ASCENDING RAMP, "D" TO DESCEND, "LINE FEED" TO STOP RAMP/<15>  
 .ASCII / USE "Q" TO START RANDOM RAMP WHERE EACH STEP IS A FRACTION OF INC VALUE/<15><1  
 .ASCII / USE C TO CYCLE ANY RAMP AT MIN,MAX VALUES; M TO CHANGE MAX. VALUE/<15><12>  
 .ASCII / SCALE RAMP INC WITH "SHIFT 2,3,4,...,0"; "SHIFT 1" TO REMOVE/<15><12>  
 .ASCII / NOTE: WAVEFORMS & NO.VOICES MAY BE SYNCH'D & RAMPED BUT CHANGE ONLY BY UNIT INC/<  
 .ASCII / OTHER LETTERS (NOT A,D,S,R,C,M,J,I,Q) RETRIEVE PRESETS; CNTRL+CHAR TO STORE/<1  
 .ASCII / PF4 (^S)=STORE PRESETS IN FILE; PF3 (^R)=RETRIEVE; PF1=LIST/<15><12>  
 .ASCII @ PF2 (^Q)=DISPLAY ENV.FRACTION/STEREO POS; CHANGE WITH ARROWS OR [POS]0<1  
 .ASCII / CNTRL A=DISPLAY AMP; CNTRL D=DECAY AT RATE=INC; CAPS LOCK OFF=DECAY/<15  
 MESS1: .ASCII / I=START RAMP FILE; J=START TRAJ.FILE (RATE=+-INC) NO.VOI, NO.VOI, TOTAL/<15  
 .ASCII / INC FREQ FRQ.RNG DUR'N DUR.RNG DELAY RAMP W.F.#2 W.F.#3 NO.VOI/

EVEN  
 SEND: MOV #4,R3 ;4 BYTES TO SEND  
 DMXWAIT  
 MOVB #4,@DMXBUF  
 20#: MOVB (R2)+,@DMXBUF  
 SOB R3,20#  
 MOVB R1,@DMXBUF  
 INC R1  
 SOB R4,SEND  
 RTS PC

INITCD: .BYTE 0,161,20,40 ;CLR ,X0,B,,S  
 .BYTE 16,161,0,40 ;CLR ,XE,B  
 .BYTE 17,161,0,40 ;CLR ,XF,B

*Fixed Waveform Grain*

CODE: NXT  
 .BYTE 25,133,300,45 ;MOVD X1,X5,BPA,D,X ;X5=SI;X=X1(SCAL) microcode  
 .BYTE 120,42,250,45 ;ADDDA X5,,,D,WY ;PHASE  
 NXT  
 .BYTE 40,42,220,46 ;ADDDA X2,,,M,S ;TABLE LOOKUP  
 .BYTE 3,173,200,45 ;MOVD ,X3,B,D ;RAMP INC  
 NXT  
 .BYTE 0,73,300,45 ;MOVD ,,,D,X ;WF VALUE TO X  
 .BYTE 60,42,250,45 ;ADDDA X3,,,D,WY ;RAMP VAL  
 .BYTE 0,73,250,105 ;MOVD ,,,D,WY,IFMI  
 NXT  
 REG: .BYTE 377,142,200,46 ;ADDDA XF,XF,B,M ;ADD TO OUTPUT  
 ENDCD: .BYTE 0,73,300,45 ;MOVD ,,,D,X ;AMP.VALUE TO X  
 .BYTE 360,72,40,40 ;MOVA XF,,,Y  
 .BYTE 340,72,40,40 ;MOVA XE,,,Y  
 .BYTE 0,73,201,46 ;MOVD ,,,M,DAC ;SCALED SAMPLE  
 .BYTE 0,73,202,46 ;MOVD ,,,M,DAC2 ;OUT  
 HALT: .BYTE 0,0,0,0

ARGPTR: 0  
 SUBROUTINE TO SCHED. EVENTS FOR DMX (20 VOICES) IN REAL TIME

TKS=177560 ;KEYBOARD REG  
 TKB=TKS+2  
 TRS=TKS+4 ;KEYBOARD OUTPUT  
 TRB=TKS+6

CONSTANTS  
 TGD=8, ;MIN.GRAIN DURATION  
 TVTX=143000 ;MAX.ADDR,  
 T01=0 ;MEMORY OFFSET PER VOICE  
 T02=4

3 W.F.MODE X1 X2 3 4 5 X6 7 X8 9 A B C D E F  
INIT: .WORD 1000,1400,0,0,0,3400,0,5400,0,0,0,0,0,0  
X1=RAMP SCALER; X2,X6,X8=WAVEFORM TABLE CENTRE ADDR.

MESS: .ASCII <12>@ Use: TAB or ARROWS to position cursor, UP/DOWN to change value@<15><12>  
.ASCII / <> to change INC; "RETURN" to print; CTRL C to exit/<15><12>  
.ASCII / SPACEBAR for new attack (at rate=INC); CTRL D for decay; '=harm./<15><12>  
.ASCII / For the first 9 variables, use these options:/<15><12>  
.ASCII / New values (or ratio) may be typed in & are put into effect with "ENTER"/<15><12>  
.ASCII / Mask direction be changed with +, - & removed with DELETE/<15><12>  
.ASCII / Halt all masks with LINE FEED/<15><12>  
.ASCII / Scale mask inc with "SHIFT 2,3,4,...,0", "SHIFT 1" to remove/<15><12>  
.ASCII / PF2 (~Q)=display env.fraction; type new vlues or use ARROWS to change/<15><12>  
.ASCII / S=start performance; P=pause (repeat to continue); C=cycle masks/<15>  
.ASCII / PF1=list; J=initiate trajectory at rate=+INC (repeat to stop)/<15><12>  
MESS1: .ASCII <12>/SEC INC PS FREQ FRQ.RNG DUR'N DUR.RNG DELAY AMP M.I. MI.RNG NO.VO  
EVEN

SEND: MOV #4,R3 ;4 BYTES TO SEND  
DMXWAIT  
MOVB #4,@DMXBUF  
20%: MOV (R2)+,@DMXBUF  
SOB R3,20%  
MOVB R1,@DMXBUF  
INC R1  
SOB R4,SEND  
RTS PC

INITCD: .BYTE 0,161,20,40 ;CLR ,X0,B,S  
.BYTE 16,161,0,40 ;CLR ,XE,B  
.BYTE 17,161,0,40 ;CLR ,XF,B

FM GRAN MICROCODE

JCODE: NXT  
.BYTE 3,173,200,45 ;MOVD ,X3,B,D ;RAMP INC  
.BYTE 63,142,210,45 ;ADDDA X3,X3,B,D,W ;RAMP VAL  
.BYTE 3,173,210,105 ;MOVD ,X3,B,D,W,IFMI  
NXT  
NXT  
.BYTE 25,133,300,45 ;MOVD X1,X5,BPA,D,X ;X5=MOD SI; X=X1  
.BYTE 120,42,250,45 ;ADDDA X5,,,D,WY ;PHASE  
NXT  
.BYTE 140,42,220,46 ;ADDDA X6,,,M,S ;TABLE LOOKUP  
.BYTE 0,73,300,45 ;MOVD ,,D,X ;M.I.  
.BYTE 60,72,40,40 ;MOVA X3,,,Y ;X RAMP  
.BYTE 0,73,300,45 ;MOVD ,,D,X ;TABLE LOOKUP  
.BYTE 0,73,240,46 ;MOVD ,,M,Y ;RAMPINDEX  
NXT  
.BYTE 5,173,200,46 ;MOVD ,X5,B,M ;HIGH ORDER  
NXT  
.BYTE 125,142,200,45 ;ADDDA X5,X5,B,D ;CAR SI  
.BYTE 4,373,200,47 ;MOVD ,X4,DB,L  
.BYTE 104,142,210,45 ;ADDDA X4,X4,B,D,W ;L.O. PHASE  
.BYTE 5,141,200,320 ;INCB ,X5,B,,,IFCS  
NXT  
.BYTE 20,72,100,40 ;MOVA X1,,,X ;SCALER  
.BYTE 120,42,250,45 ;ADDDA X5,,,D,WY ;H.O. PHASE  
.BYTE 0,60,0,40 ;NOP  
.BYTE 40,42,220,46 ;ADDDA X2,,,M,S ;TABLE LOOKUP  
.BYTE 60,72,40,40 ;MOVA X3,,,Y ;X RAMP  
.BYTE 0,73,300,45 ;MOVD ,,D,X  
NXT

REG: .BYTE 377,142,200,46 ;ADDDA XF,XF,B,M ;ADD TO OUTPUT

ENDCD: .BYTE 0,73,300,45 ;MOVD ,,D,X ;AMP.VALUE  
.BYTE 360,72,40,40 ;MOVA XF,,,Y  
.BYTE 340,72,40,40 ;MOVA XE,,,Y  
.BYTE 0,73,201,46 ;MOVD ,,M,DAC  
.BYTE 0,73,202,46 ;MOVD ,,M,DAC2  
HALT: .BYTE 0,0,0,0  
EVEN

ARGPTR: 0

SUBROUTINE TO SCHED. F.M. EVENTS FOR DMX (8 VOICES) IN REAL TIME

KS=177560 ;KEYBOARD REG  
KB=TKS+2 ;KEYBOARD OUTPUT  
RS=TKS+4