SUPERSONI95
DARKY
INSTRUCTION
MANUAL

CONTENTS

PROLOGUE

Welcome to the Darky universe. Darky is our most complex product yet and it took us 6 years to develop. We are glad you are diving into this adventure with us. Darky is different. Out of the ordinary. Alien to our world of normal PSG sound.

Darky's journey started in January 2014 when we got the idea to create something new but still familiar. We were pitching some new ideas for new MSX hardware and after creating two cartridges with support for the Sega Master System with a VDP that's somewhat compatible with the MSX1 VDP (our Franky and PlaySoniq products) we were looking for other hardware that might work out of the box on MSX, but with some extra modes and features.

And then we found the AY8930 sound processor, called the ePSG. It's backward compatible with our beloved AY-3-8910 processor, but with extra modes with a lot of extra possibilities to create unique PSG sounds. It was created by General Instruments, just before it was sold to Microchip.

So we decided to build a ePSG cartridge. But why not a stereo ePSG cartridge? There's a lot of 6 channel PSG tracks now from arcade and other computer systems. And then we thought that it would also be nice if all 6 channels could be controlled- individually so they can freely be placed in the stereo spectrum, by adjusting the left and right volume. So we added two 6-channel stereo mixers to support this. One stereo channel per PSG channel.

But we didn't stop there. To add some more flavor to the sound output, we looked and found an effect processor we could use. The Spin SPN1001 FV-1 to be exact. The FV-1 processor has 8 fixed effects, but with an additional EEprom we could increase this with an extra 8 effects. These last 8 effects can be programmed and uploaded by you. There are some programs on PC that can help you create those effects. We created the upload tooling for Darky. After building some prototypes developers asked us two things: "would it be possible to bypass the effect processor?" and: "could you make it so that we can add other sound sources to the effect processor?". We decided to add this support, but the changes in the hardware design took us another two years to develop, build and test.

And then we decided it would also be cool if you could control the soundcard in real-time. For this we added a special port on the Darky soundcard, that enables remote control of all the functions available. As we are writing this, this external controller is in the final stages of development.

We had many challenges to overcome during the development of Darky. Thank you for supporting us. We hope you enjoy using the Darky sound card as much as we do.

The SuperSoniqs Team

ACKNOWLEDGEMENTS

OPERATING INSTRUCTIONS

Please power off your MSX before inserting or removing the Darky sound card. Darky consists of more than 250 parts and a lot of those are sensitive.

STANDARD OPERATING MODE

In standard operating mode Darky will serve as a "normal" PSG playback device, with a second PSG activated on a slightly different clock frequency to create a stereo effect for 3 channel PSG music. It will work with all your existing games and programs. By setting register values you can change this behavior so you can play 6-channel PSG tracks, or you can set Darky into the ePSG mode, to get access to more registers.

FUNCTIONALITY

Darky has many functions. You can start by just using Darky as your preferred PSG playback device, and progress to explore her many functions step by step. Basically, only your own creativity and the time you spend with Darky will decide how many possibilities you'll unlock.

Two ePSG Processors

Darky has two AY8930 ePSG processors. Apart from being fully backwards compatible with the AY-3-8910, the AY8930 has an expanded mode.

This expanded mode has the following features:
- separate envelopes for each channel (as opposed to one shared envelope in the AY-3-8910)
- Improved frequency range
- more bits of precision for note frequency
- Separate frequency, duty cycle and envelope controls for each channel
- 5 Bits of Logarithmic digital to analog conversion per channel

Please see our Appendix for the register information of the AY8930 (e)PSG modes. For further study, please refer to the AY8930 Technical Datasheet.

Spin FV-1 FX Processor

The Spin FV-1 Effect Processor has 8 build in programs, and 8 programmable slots. There is various software for Mac and PC to program your own effects and the compiled results can be uploaded into Darky to use in your own programs. Various repositories with pre-build effects exist, you can download effects from those sources and upload them also. We hope of course that in the future programs will be made to create effects on the MSX itself.

Mark Stratman keeps a effect repository at github: https://mstratman.github.io/fv1-programs

For information about the Java programming environment for the Spin FV-1 please look at the ElmGen repository at github. https://github.com/hires/ElmGen

Spincad Designer is a visual programming tool based on ElmGen. The github repository is here: https://github.com/HolyCityAudio/SpinCAD-Designer

For information about the official Windows assembler SpinASM please look at the SpinASM website: http://www.spinsemi.com/knowledge_base/pgm_quick.html of Spin Semiconductor and the manual: http://www.spinsemi.com/Products/datasheets/spn1001-dev/SPINAsmUserManual.pdf

Two PT2322 Mixers

Darky has two 6-channel PT2322 mixers. Every single ePSG output channel is connected to a left and right stereo input channel on the PT2322. The PT2322 supports settings like (master) volume adjusting, equalizing (treble, middle, and bass), 3D effect functions, tone defeat and mute. Because volume can be balanced per ePSG channel, every ePSG channel can be set differently in the stereo spectrum (e.g. setting left 50% of volume compared to right on a specific ePSG output channel, means that particular ePSG output channel will sound more to the left in the stereo spectrum. This can be done for each of the 6 ePSG output channels. And it can be adjusted in real-time also in your own programs or with our external controller (in development).

MSX-DOS COMMANDS

Darky comes with its own set of MSX-DOS commands. These are:

DKFXLOAD <name.bin>

With DKFXLOAD you can load up to 8 extra effects for the Spin FV-1 Effect processor in Darky's dedicated EEPROM. The file needs to be a compiled binary according to the Spin Assembler output specifications. After executing, the command will ask to with slot (1-8) you want to load the effect. DKFXLOAD will see files of 512 bytes or smaller as one single effect and it will ask you in which slot you want to load the effect. When the biniary file size is equal to 4096 bytes, DKFXLOAD assumes that file contains information for all 8 free slots and will load that data into all 8 slots.

DKFXSAVE <name.bin>

With DKFXSAVE you can save effects from one of the eight extra effect banks to your writable media. By default the file will be saved in the directory you have started this command. After executing, the command will ask from which slot (1-8, or A for all slots) you want to save the effect data.

DKREG <REG NUMBER>;<Reg Value>

With DKREG you can access the internal registers of Darky. As a distinction between the two parameters, you can use either a <;> or a <,>.

Register readout:
dkreg <Reg number>
dkreg <Reg number>?
dkreg ?<Reg number>;<Read value>
dkreg <Reg number>;<Read value>?

Example:
dkreg ?0x70;0 reads internal register 0.

It doesn't matter where you place the question mark, when dkreg encounters one in the line entered it will read out a register. You can use both decimal and hexadecimal notations, hex values need to be written with 0x instead of &H. Reading back must always be done through register 0x60, so for example DKREG ?0x60,0 will read back internal register 0.

DKUPDATE <name>

Tool for updating the Darky firmware.

DARKY SLOT SWITCHING

It's possible to use two Darky soundcards in one MSX. To be able to use both, you need to change the base address of one of the Darky sound cards.

1-Check if you can detect a Darky on the +1 address, if so initializing was already done and you are ready
2-Enable slot detect bit (0x41 bit 3 'EN_SLOT_DET')
3-Reset current +1 address configuration by setting 'RES_SLOT_DET' bit (0x41 bit 4), reset afterwards
4-Read page 1 in the slot where you want to check if Darky is present
5-Check if Darky is present on switched I/O +1
6-If not, switch to next slot and repeat step 4. Keep repeating until all slots are done or a second Darky is found.
7-Disable slot detect
8-If in the end you are left with only a Darky at the +1 address, then there was only one Darky inserted, you can reset the +1 configuration again by using the 'RES_SLOT_DET' bit.

Overall don't read from Darky before you completed the +1 address shift! Also: for your detecting routine it's better to assume there are two Darky's inserted and look for them both, to prevent possible short circuiting of the MSX data bus, which will give unexpected results of operation.

Please study the following example (for primary slots):

```
char DK_SearchSlot(SlotNr)
char SlotNr;
{
        #asm
                        LD HL,2                 ;Get stack pointer into HL and skip ret address
                        ADD HL,SP
                        LD A,(HL)               ;Get slot number
                        SLA A                   ;Shift value to bit 2&3
                        SLA A
                        AND 00CH
                        LD B,A                  ;Store new slot number

                        LD HL,0                 ;Default return non-present state (0)

                        LD A,0AAH               ;Enable darky on default switched IO address
                        OUT (040h),A

                        DI                      ;Disable IRQ
                        LD A,008H               ;Enable slot detect
                        OUT (041h),A

                        IN A,(0A8H)             ;Get current slot settings
                        LD C,A
                        AND 0F3H
                        OR B                    ;Set new slot settings
                        OUT (0A8H),A
                        LD A,(04000H)           ;Read an address inside page 1
                        LD A,C                  ;Restore slot settings
                        OUT (0A8H),A
```

```
                    SUB A                   ;Disable slot detect
                    OUT (041h),A
                    EI                      ;Enable IRQ

                    LD A,0ABH               ;Enable darky on increased switched IO address.
                    OUT (040h),A
                    IN A,(040h)             ;Read switched IO value
                    CP 054H      ;Check if invert ID is present inside switched I/O
                    RET NZ
                    LD L,1       ;Return 1 if darky was found
        #endasm
```

DARKY REGISTER SET
Switched I/O ID
Switched I/O ID = 170 (AAh) for Darky (Supersoniqs)
Switched I/O ID = 171 (ABh) for slot increased Darky

I/O Addresses
40H W: Using 170 as ID to enable darky switched I/O 41H..4FH
41H RW: CPLD control register
42H RW: Internal register (b7='1'-address and b7='0'-Data)
44H W: PSG 1 Index
45H W: PSG 1 Data write
46H R: PSG 1 Data read
4CH W: PSG 2 Index
4DH W: PSG 2 Data write
4EH R: PSG 2 Data read

CPLD Control register timing
-Writing index to port 42H (Bit 7='1') takes 2uS
-Writing data to port 42H (Bit 7='0') takes 7uS
-Writing data to active BUSY status takes 2uS

CPLD Control register 41H
Bit 0-R-PSG 1 Mirror to A0/A1, '0'=off/'1'=on (Default)
Bit 1-R-PSG 2 Mirror to A0/A1, '0'=off/'1'=on (Default)
Bit 2-R-PSG 2 Clk select, '0'=3.579545MHz/'1'=3.6MHz (Default)
Bit 3-R-CPLD Version bit 0 (Read only)
     W-EN_SLOT_DET (Write only), enables possibility to change switched I/O address to +1)
        (When enabled, reading/writing from page1 on the slot where darky is in, will
        increase address)
Bit 4-R-CPLD Version bit 1 (Read only)
     W-RES_SLOT_DET (Write only), only resets if EN_SLOT_DET is enabled)
Bit 5-R-CPLD Version bit 2 (Read only)
Bit 6-R-Command pending (1=BUSY)
Bit 7-R-CPLD &H42 write toggle

42H Internal register set
PT2322 Left
0x00-* W-Panning left channel 1 (PSG1-A)(0-15 (0dB to -15dB))
0x01-* W-Panning left channel 2 (PSG1-B)
0x02-* W-Panning left channel 3 (PSG1-C)
0x03-* W-Panning left channel 4 (PSG2-A)
0x04-* W-Panning left channel 5 (PSG2-B)
0x05-* W-Panning left channel 6 (PSG2-C)
0x06-* W-Function select XXXX321X
         bit 1-0=Tone Control ON   1=TONE Defeat
         bit 2-0=3D On   1=3D Off
         bit 3-0=Mute Off   1=Mute On
0x07-* W-Bass tone control left   (0-15) (0=-14dB, 7=0dB, 8=+14dB,15=-0B)
0x08-* W-Middle tone control left (0-15) (0=-14dB, 7=0dB, 8=+14dB,15=-0B)
0x09-* W-Treble tone control left (0-15) (0=-14dB, 7=0dB, 8=+14dB,15=-0B)
0x0A-* W-Master volume (Bit 4,5,6=-10dB steps, Bit 0,1,2,3=-1dB steps)

0x0B-* W-Linear master volume 0 to -63dB in 1dB steps. This Linear volume register is controlling the PT2322 the same as the master volume register but easier to use for programmers as it's just one single 0-63 decimal value you use instead of two dB ranges.

PT2322 Right
0x10-* W-Panning Right channel 1 (PSG1-A)
0x11-* W-Panning Right channel 2 (PSG1-B)
0x12-* W-Panning Right channel 3 (PSG1-C)
0x13-* W-Panning Right channel 4 (PSG2-A)
0x14-* W-Panning Right channel 5 (PSG2-B)
0x15-* W-Panning Right channel 6 (PSG2-C)
0x16-* W-Function select XXXX321X
          bit 1-0=Tone Control ON 1=TONE Defeat
          bit 2-0=3D On   1=3D Off
          bit 3-0=Mute Off   1=Mute On
0x17-* W-Bass tone control Right   (0-15) (0=-14dB, 7=0dB, 8=+14dB,15=+0B)
0x18-* W-Middle tone control Right Right (0-15) (0=-14dB, 7=0dB, 8=+14dB,15=-0B)
0x19-* W-Treble tone control Right (0-15) (0=-14dB, 7=0dB, 8=+14dB,15=-0B)
0x1A-* W-Master volume (Bit 4,5,6=-10dB steps (0-7), Bit 0,1,2,3=-1dB steps (0-9)),value 0dB=Max volume/-79dB lowest volume
0x1B-* W-Linear master volume 0 to -63dB in 1dB steps. This Linear volume register is controlling the PT2322 the same as the master volume register but easier to use for programmers as it's just one single 0-63 decimal value you use instead of two dB ranges.

Spin FV-1 processor
0x20-* W-Trim pot 1 (0x00-0x1F)
0x21-* W-Trim pot 2 (0x00-0x1F)
0x22-* W-Trim pot 3 (0x00-0x1F)
0x24-* W-Program mode Bit 0,1,2=Program mode (0-7), Bit 3=Int/Ext program mode (1=External)

Prg	Description	POT0	POT1	POT2
0	Chorus-reverb	Reverb mix	Chorus rate	Chorus mix
1	Flange-reverb	Reverb mix	Flange rate	Flange mix
2	Tremolo-reverb	Reverb mix	Tremolo rate	Tremolo mix
3	Pitch shift	Pitch +/-4 semitones	-	-
4	Pitch-echo	Pitch shift	Echo delay	Echo mix
5	Test	-	-	-
6	Reverb 1	Reverb time	HF filter	LF filter
7	Reverb 2	Reverb time	HF filter	LF filter

0x28-R-Read 24LC32A Status register (Bit 0-'1'=Memory transfer request busy)
0x29-W-Write 24LC32A Index value (0-31)
0x2A-R-Read 24LC32A buffer from index location, Index is auto increment
0x2B-W-Write 24LC32A lower data nibble
0x2C-W-Write 24LC32A higher data nibble and write data byte to buffer at index location, Index is auto increment
0x2D-W-Start memory read transfer from 24LC32A to buffer (Page number 0-127)
0x2E-W-Start memory write transfer from buffer to 24LC32A (Page number 0-127)

Mixer Control

Please study appendix A to understand the audio flow within Darky.
(Mixer volume level settings for initially adjusting only, will tick/pop when changed!)
0x30-* W-Audio left in mixer volume/balance channel (0(mute)-31)
0x31-* W-Audio right in mixer volume/balance channel(0(mute)-31)
0x32-* W-Audio mixer routing
        Bit 0-PSG FX Bypass (Default '0')
        Bit 1-External line-in FX Bypass (Default '0')
        Bit 2-External line in audio select ('0'=MSX bus audio/'1'=Line in audio)

CPLD Control

0x40-* W-CPLD control register write
        Bit 0-RW-PSG 1 Mirror to A0/A1 (Default on, No read) (Mute PSG before changing)
        Bit 1-RW-PSG 2 Mirror to A0/A1 (Default on, No read)-(Mute PSG before changing)
        Bit 2-RW-PSG 2 Clk select, '0'=3.579545MHz/'1'=3.6MHz

Parameter set save/read control
0x44-* W-Select set number 0-7 (Set 0 will load after power-up)
0x45-* W-Write current settings to non volatile memory set (Write '1' to this register to
activate writing, will read zero afterwards)
0x46-* W-Read settings set from non volatile memory (Write '1' to this register to activate
writing, will read zero afterwards)

External control registers

0x50-* W-Enable external controller ('1'-on)

Readback Darky registers

for all above registers marked with an *
0x60- R -Readback specific Darky register (Data value is internal register number,
        value can be read after command is ready)

Darky specific

0x70- R -Internal operations status register
        Bit 0-Left PT2322 data transfer status ('0'=Busy/'1'=Ready)
        Bit 1-Right PT2322 data transfer status ('0'=Busy/'1'=Ready)
        Bit 2-Audio in balance MCP4651 data transfer status ('0'=Busy/'1'=Ready)
        Bit 3-Non volatile memory CRC error
        Bit 4-External power detect
        Bit 5-External power shorted
        Bit 6-External control timeout error
        Bit 7-External control NAK error
0x71- R -Get highest external control state number (Resets after read)
0x72-R -Get external control error counter (Resets after read)
0x77-R -Read back page or index number written into register 0x78
0x78- W-Set page number (0-0x7F) and reset offset when in bootloader mode
        -Set index number (0-0x7F) when in application mode
0x79- W-Write low data nibble when in bootloader mode (Auto register index increase)
0x7A- W-Write high data nibble and transfer data to page when in bootloader mode
0x7B-RW-Loop written data (0-0x7F) to read (For test purposes only, both modes)
0x7C-R -Recheck application CRC when in bootloader mode and readback CRC status (0=wrong/1=ok)
0x7D-R -Version XX.YY (Bit 7..4=X/Bit3..0=Y) Max value=15.15
0x7E- W-Start application mode when in bootloader mode (Page should be 07Fh)
        -Start bootloader mode when in application mode (Index should be 07Eh)
0x7F-R -Mode (0=Application/1=Bootloader)

Wait for TurboR Machines

Keep in mind that after initiating a internal register read, you always have to wait for
completion by polling the busy bit. (N.B. this is only for reading, not for writing). The Turbo-
R in turbo mode is by default too fast to write Darky register &H42. After writing an index
value always wait 2uS, after writing a data value always wait 7uS. Between writing data and
reading the busy bit, there should always be a waiting time of 2uS.

```
        Read & write example:

ReadReg:
                LD          A,060h                  ;Set readback register
                OR          080h
                CALL        WriteIndex
                LD          A,<darky register number you want to read>
                AND         07Fh
                CALL        WriteData               ;Start readback command
                CALL        WaitBusy                ;Wait for read command to complete
                IN          A,(042h)                ;Read darky readback register
                RET

WriteIndex:
                OUT         (042h),A                ;Write new index to Darky
                EX          (SP),IX                 ;Addition wait for Turbo-R machines
                EX          (SP),IX
                RET
WriteData:
                OUT         (042h),A                ;Write new data to Darky
                EX          (SP),IX                 ;Addition wait for Turbo-R machines
                EX          (SP),IX
                EX          (SP),IX
                EX          (SP),IX
                RET

WaitBusy:
                EX          (SP),IX                 ;Addition wait for Turbo-R machines
                EX          (SP),IX
                IN          A,(041h)                ;Get CPLD control register
                AND         040h                    ;Check busy bit
                JR          NZ,WaitBusy             ;Loop if darky is still busy
                RET
```

Quadraphonic sounds

Darky now supports another Darky.

An additional flipflop is implemented that can be reset by the CPLD control register and set by reading somewhere in page 1 at a specific slot.

As it's set it will increase the switched i/o ID thereby enabling that specific Darky on a different id.

Switched I/O addresses:

170   Darky

171   Second Darky if available

Together with the internal MSX PSG a total of 15 PSG channels can then be used.

DARKY TEST PROGRAM

The following program made by Laurens Holst can be used to test your Darky sound card (please note that this program is also included in the downloadable Darky archive on our website). Pleas study and play with it if you want to try to test things under MSX basic. For trying settings under MSX-DOS you can use our DKREG.COM program.

```
10 DEFINT A-Z
20 PRINT "Detecting Darky: ";
30 OUT &H40,170:A=INP(&H40)   ` Select Darky expanded I/O bank
40 IF A=85 THEN PRINT "Found" ELSE PRINT "Not found":END
50 GOSUB 110  ` PSG tests
60 GOSUB 170  ` Mixer tests
70 GOSUB 260  ` Spin tests
80 OUT &H42,&HC0:OUT &H42,&B111   ` Enable A0 mirror & detune
90 END
100 ` PSG tests
110 ` PRINT "Detecting PSG 1: ";
```

```
120 C=&H44:GOSUB 350
130 PRINT "Detecting PSG 2: ";
140 C=&H4C:GOSUB 350
150 RETURN
160 ` Mixer tests
170 PRINT "Mixer ready check: ";
180 OUT &H42,&HF0:OUT &H42,0:A=INP(&H42) AND 7 XOR 7
190 IF A THEN PRINT "Failed" ELSE PRINT "Done"
200 PRINT "Mixer sound test: ";
210 GOSUB 740
220 PRINT "Line in mixer sound test: ";
230 GOSUB 580
240 RETURN
250 ` Spin tests
260 PRINT "Spin FV-1 internal sound test: ";
270 P=4:P0=0:P1=31:P2=31:GOSUB 900
280 PRINT "Spin FV-1 external load: ";
290 GOSUB 1040
300 PRINT "Spin FV-1 external verify: ";
310 GOSUB 1170
320 PRINT "Spin FV-1 external sound test: ";
330 P=8:P0=15:P1=7:P2=18:GOSUB 900
340 RETURN
350 ` PSG test subroutine
360 OUT &H42,&HC0:OUT &H42,&B000  ` Disable A0 mirror & detune
370 OUT C,13:OUT C+1,0  ` Select AY-3-8910 compatibility mode
380 OUT C,0:OUT C+1,&H55:A=INP(C+2)  ` Test if r#0 is writable
390 IF A<>&H55 THEN PRINT "Not found":RETURN
400 OUT C,6:OUT C+1,&HFF:A=INP(C+2)  ` Test noise period mask
410 IF A<>&H1F THEN PRINT "Yamaha YM2149":RETURN
420 OUT C,13:OUT C+1,&HA0  ` Select AY8930 mode bank A
430 OUT C,6:OUT C+1,&H55:A=INP(C+2)  ` Test noise period mask
440 IF A=&H55 THEN PRINT "Microchip AY8930" ELSE PRINT "GI AY-3-8910":RETURN
450 PRINT "AY8930 sound test: ";
460 OUT C,0:OUT C+1,0:OUT C,1:OUT C+1,2  ` Set tone A period
470 OUT C,7:OUT C+1,&B10111110:OUT C,8:OUT C+1,31  ` Enable channel A
480 OUT C,13:OUT C+1,&HB0  ` Select AY8930 mode bank B
490 FOR I=0 TO 8
500 T=TIME
510 OUT C,6:OUT C+1,I  ` Set pulse width
520 IF TIME-T<20 GOTO 520
530 NEXT
540 OUT C,13:OUT C+1,0  ` Select AY-3-8910 compatibility mode (reset)
550 PRINT "Done"
560 RETURN
570 ` Line in mixer test subroutine
580 OUT &H42,&HC0:OUT &H42,&B000  ` Disable EPSG A0 mirror
590 OUT &H42,&HB0:OUT &H42,0
600 OUT &H42,&HB1:OUT &H42,0
610 OUT &H42,&HB2:OUT &H42,&B111
620 FOR I=0 TO 93
630 T=TIME
640 IF I<63 THEN OUT &H42,&HB0:OUT &H42,31-ABS(I-31)
650 IF I>31 THEN OUT &H42,&HB1:OUT &H42,31-ABS(I-62)
660 IF TIME-T<4 GOTO 660
670 NEXT
680 OUT &H42,&HB0:OUT &H42,0
690 OUT &H42,&HB1:OUT &H42,0
700 OUT &H42,&HB2:OUT &H42,&B100
710 PRINT "Done"
720 RETURN
730 ` PT2322 mixer test subroutine
740 OUT &H42,&HC0:OUT &H42,&B001  ` Enable EPSG 1 A0 mirror
750 OUT &H42,&H8A:OUT &H42,&H79  ` Min master volume (left)
760 OUT &H42,&H9A:OUT &H42,&H79  ` Min master volume (right)
770 PLAY "t32v15a1"
780 FOR I=0 TO 237
790 T=TIME
800 IF I<=158 THEN OUT &H42,&H8A:J=ABS(I-79):OUT &H42,(J\10)*16+(JMOD10)
810 IF I>=79 THEN OUT &H42,&H9A:J=ABS(I-158):OUT &H42,(J\10)*16+(JMOD10)
820 IF TIME-T<2 GOTO 820
```

```
830 NEXT
840 IF PLAY(0)<>0 GOTO 840
850 OUT &H42,&H8A:OUT &H42,0   ` Max master volume (left)
860 OUT &H42,&H9A:OUT &H42,0   ` Max master volume (right)
870 PRINT "Done"
880 RETURN
890 ` Spin FV-1 test subroutine
900 OUT &H42,&HC0:OUT &H42,&B001   ` Enable EPSG 1 A0 mirror
910 OUT &H42,&HA4:OUT &H42,P   ` Program
920 OUT &H42,&HA0:OUT &H42,P0  ` Pot 0
930 OUT &H42,&HA1:OUT &H42,P1  ` Pot 1
940 OUT &H42,&HA2:OUT &H42,P2  ` Pot 2
950 PLAY "t120v15l8arbrcrarbrcr"
960 IF PLAY(0)<>0 GOTO 960
970 OUT &H42,&HA4:OUT &H42,5
980 OUT &H42,&HA0:OUT &H42,0
990 OUT &H42,&HA1:OUT &H42,0
1000 OUT &H42,&HA2:OUT &H42,0
1010 PRINT "Done"
1020 RETURN
1030 ` Load Spin FV-1 external program 0
1040 RESTORE 1330
1050 FOR I=0 TO 15
1060 GOSUB 1300:OUT &H42,&HA9:OUT &H42,0
1070 READ A$:PRINT HEX$(I);CHR$(29);
1080 FOR J=1 TO 64 STEP 2
1090 GOSUB 1300:OUT &H42,&HAB:OUT &H42,VAL("&H"+MID$(A$,J+1,1))
1100 GOSUB 1300:OUT &H42,&HAC:OUT &H42,VAL("&H"+MID$(A$,J,1))
1110 NEXT
1120 GOSUB 1300:OUT &H42,&HAE:OUT &H42,I
1130 NEXT
1140 PRINT "Done"
1150 RETURN
1160 ` Verify Spin FV-1 external program 0
1170 RESTORE 1330
1180 FOR I=0 TO 15
1190 GOSUB 1300:OUT &H42,&HA9:OUT &H42,0
1200 GOSUB 1300:OUT &H42,&HAD:OUT &H42,I
1210 READ A$:PRINT HEX$(I);CHR$(29);
1220 FOR J=1 TO 64 STEP 2
1230 GOSUB 1300:OUT &H42,&HAA:OUT &H42,0:A=INP(&H42):B=VAL("&H"+MID$(A$,J,2))
1240 IF A<>B THEN PRINT "Failed":RETURN
1250 NEXT
1260 NEXT
1270 PRINT "Done"
1280 RETURN
1290 ` Wait Spin FV-1 external memory access
1300 OUT &H42,&HA8:OUT &H42,0:IF INP(&H42) AND 1 GOTO 1300
1310 RETURN
1320 ` Sine generator DSP program
1330 DATA 804000110000400D000004264000022400A4FF6D280002044000AB8D4000000C
1340 DATA 00A404654000004660000040063C00024400A404854000004864000880D4000000C
1350 DATA 000004A6400000444000040A40000424C00004260000040A4000044400000446
1360 DATA 40000444000004040A40000424C00004260000040A4000044400000446400000444
1370 DATA 0000040A40000424C00004260000040A4000044400000446400000444000040A
1380 DATA 40000424C00004260000040A400004447F5C0446000004AA400002C6000002E6
1390 DATA 0000001100000011000000110000001100000011000000110000001100000011
1400 DATA 0000001100000011000000110000001100000011000000110000001100000011
1410 DATA 0000001100000011000000110000001100000011000000110000001100000011
1420 DATA 0000001100000011000000110000001100000011000000110000001100000011
1430 DATA 0000001100000011000000110000001100000011000000110000001100000011
1440 DATA 0000001100000011000000110000001100000011000000110000001100000011
1450 DATA 0000001100000011000000110000001100000011000000110000001100000011
1460 DATA 0000001100000011000000110000001100000011000000110000001100000011
1470 DATA 0000001100000011000000110000001100000011000000110000001100000011
1480 DATA 0000001100000011000000110000001100000011000000110000001100000011
```

APPENDIX A: MIXER FLOW AND MINI JACK WIRING
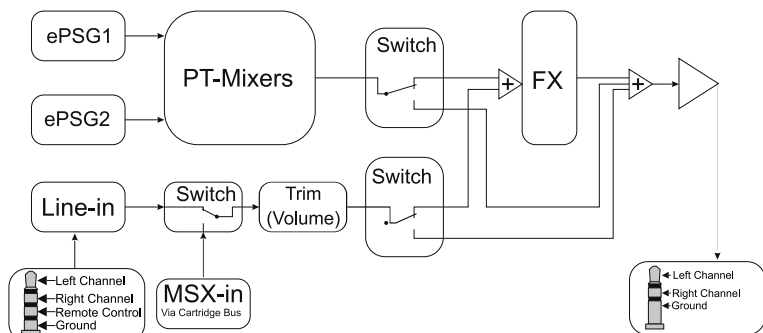
Please note that the top left mini jack on your Darky is the stereo output. As shown in the above picture on the right. The mini jack in the middle is the 4-pin audio and external control input. Please wire your cable as in the example above on the lower left. Please note that inserting a normal stereo mini jack into the 4-pin mini jack will cause Darky to block that port as a precaution until you remove that cable.

APPENDIX B: AY8930 REGISTER INFORMATION

Excerpt taken from the AY8930 Technical Datasheet. Please refer to the AY8930 datasheet for the complete information.

Register Array

The basis of the AY8930 is an array of 27 control registers arranged in one bank of 16 and one bank of 11 registers. These registers occupy 16 address locations of the 1,024-word memory space in which the PSG resides.

The configuration of this register array is shown in the following tables. Note the two modes of operation: 8910A-compatibility mode and 8930 expanded mode.

The registers are addressed via the combination of the bidirectional data bus (DA0-DA7) and address input pins A8 and A9.

A9	A8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
0	1	0	0	0	0	X	X	X	X

DA0-DA3 These four low-order address bits are used to select one of the internal registers within a bank. DA4-DA, A8, A9 These six high-order address bits function as chip selects and are used to position the register bank(s) within the 1,024-word memory space. In the deselected state, the data bus is in the high impedance state. The address enable code for bits DA4-DA7 is all zeros. Inputs A8 and A9 are enabled

14

by a high on A8 and a low on A9; all other input level combinations result in a deselected condition. Pins A8 and A9 have an on-chip pull-up and pulldown resistor, respectively, and will assume the correct logic level if left unconnected.

Address bits DA7-DA0 are latched internally. This internally latched address is updated and modified on every "latch address" signal presented to the PSG via the BDIR and BC1 control lines. The AY8930 initializes in the AY38910A-compatibility mode. To utilize the expanded features of the AY8930, an access code must be input to register R15 upon program initialization. Entering a "101" code in bits B7-B5 of register R15 selects the 8930 expanded mode. In the 8930 expanded mode, bit B4=0 (R15) selects BANKA and B4 = 1 selects BANK B. All other bit selections are defined as 8910A- compatibility mode. Registers R15A and R15B are mapped into the same physical register.

! Switching modes causes loss of all register data from the previous mode. All registers will be initialized except for the Mode Select code of R1S.

Shown on the next page is the register configuration for the AY8930. Note that Bank A of the expanded mode is virtually identical to the single register array of the 8910A-compatibility mode.

AY8930 Register Array AY8910a-Compatibility Mode

Register HEX	Register Octal	Function	B7	B6	B5	B4	B3	B2	B1	B0
R0	R0	Channel A Tone Period	8-Bit Fine Tune							
R1	R1						4-Bit Coarse Tune			
R2	R2	Channel B Tone Period	8-Bit Fine Tune							
R3	R3						4-Bit Coarse Tune			
R4	R4	Channel C Tone Period	8-Bit Fine Tune							
R5	R5						4-Bit Coarse Tune			
R6	R6	Noise Period				5-Bit Period Control				
R7	R7	Enable	IN/OUT IOB	IOA	Noise C	B	A	Tone C	B	A
R8	R10	Channel A Amplitude				M	L3	L2	L1	L0
R9	R11	Channel B Amplitude				M	L3	L2	L1	L0
RA	R12	Channel C Amplitude				M	L3	L2	L1	L0
RB	R13	Envelope Period	8-Bit Fine Tune							
RC	R14		8-Bit Coarse Tune							
RD	R15	Envelope Shape/ Cycle	MODE SELECT				CONT	ATT	ALT	HOLD
RE	R16	I/O Port A	8-Bit Parallel I/O on Port A							
RF	R17	I/O Port B	8-Bit Parallel I/O on Port B							

AY8930 Register Array Expanded Capability Mode - Bank A

Register			Bit							
HEX	**Octal**	**Function**	**B7**	**B6**	**B5**	**B4**	**B3**	**B2**	**B1**	**B0**
R0A	R0A	Channel A					8-Bit Fine Tune			
R1A	R1A	Tone Period					8-Bit Coarse Tune			
R2A	R2A	Channel B					8-Bit Fine Tune			
R3A	R3A	Tone Period					8-Bit Coarse Tune			
R4A	R4A	Channel C					8-Bit Fine Tune			
R5A	R5A	Tone Period					8-Bit Coarse Tune			
R6A	R6A	Noise Period					8-Bit Noise Period			
			IN/OUT		**Noise**			**Tone**		
R7A	R7A	$\overline{\text{Enable}}$	IOB	IOA	C	B	A	C	B	A
R8A	R10A	Channel A Amplitude			M	L4	L3	L2	L1	L0
R9A	R11A	Channel B Amplitude			M	L4	L3	L2	L1	L0
RAA	R12A	Channel C Amplitude			M	L4	L3	L2	L1	L0
RBA	R13A	Channel A					8-Bit Fine Tune			
RCA	R14A	Envelope Period					8-Bit Coarse Tune			
RDA	R15A	Bank A/B: Envelope A	1	0	1	0	CONT.	ATT.	ALT.	HOLD
REA	R16A	I/O Port A	8-Bit Parallel I/O on Port A							
RFA	R17A	I/O Port B	8-Bit Parallel I/O on Port B							

AY8930 Register Array Expanded Capability Mode - Bank B

HEX	Octal	Function	B7	B6	B5	B4	B3	B2	B1	B0
Register			**Bit**							
R0B	R0B	Channel B	8-Bit Fine Tune							
R1B	R1B	Envelope Period	8-Bit Coarse Tune							
R2B	R2B	Channel C	8-Bit Fine Tune							
R3B	R3B	Envelope Period	8-Bit Coarse Tune							
R4B	R4B	Envelope Shape/Cycle B					CONT.	ATT	ALT	HOLD
R5B	R5B	Envelope Shape/Cycle C					CONT.	ATT	ALT	HOLD
R6B	R6B	Channel A Duty Cycle					4-Bit			
R7B	R7B	Channel b Duty Cycle					4-Bit			
R8B	R10B	Channel c Duty Cycle					4-Bit			
R9B	R11B	Noise "And" Mask	8-Bit							
RAB	R12B	Noise "Or" Mask	8-Bit							
RBB	R13B*									
RBC	R14B									
RBD	R15B	Bank A/B: Envelope A	1	0	1	1	CONT.	ATT.	ALT.	HOLD
RBC	R16B*									
RBF	R17B*	TEST	NOT TO BE ACCESSED BY THE USER							

18

NOTES

DARKY
INSTRUCTION
MANUAL