# MIDIBox SNES APU

## What is it?

### Rough Overview

The SNES-APU (or **S**uper **N**intendo **E**ntertainment **S**ystem - **A**udio **P**rocessing **U**nit) is the module inside the Super Nintendo that handled all the processing and direct playing of music and effects. It consists of a processor (the Sony SPC700), and a DSP all rolled into a stand-alone module that plugs into the SNES via a 24 pin connector.

### This Page

This page is where all work and all useful notes and correspondance will be posted. This will hopefully generate interest in the project and help to get it off the ground faster. Photos, links and data papers will be posted in order to amass what is needed to make a PCB as well as software.

## Existing Projects

Much is already known about the unit and there is a mixture of projects that have already been successful with the unit. Projects such as connecting a APU via parallel port to a PC and of course the really nitty gritty work of emulating the entire behaviour of the APU itself.

In short there are two distinct things that have been done with the SNES APU:

### Emulation: Play SPC songs

There are numerous individuals who have been able to load SPC files onto a APU connected to their PC's Parallel port. The SPC is a emulation of a the song/program that is from a SNES game. This work proves that is is possible to harness the APU outside of the SNES. However, songs or programs have not been written for these parallel/APU setups.

Many have open source programs written in different languages such as c++ and vBasic. One of the most prepared project is below where .dll and .c files that work with a APU plugged into a parallel port are provided. Links to Windows and Linux drivers and applications are provided. This person even has a wishlist of having a computer-separate unit like the SIDplayer!

http://www.raphnet.net/electronique/snes_apu/snes_apu_en.php#schematic

### Homebrew: Create SPC songs

Numerous homebrew SNES gamemakers have been able to follow the copious third-party

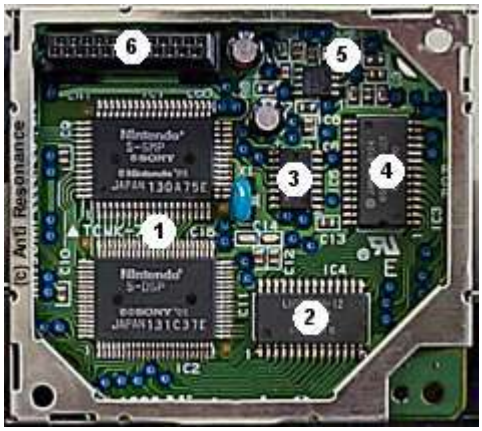
documentation that exists for the APU and have created music for the SNES that function appropriately via emulation (as most homebrew is not burned to cartridges and never really “make-it” into a APU).

### So...

Basically what we're looking at is a marriage of the existing two things into one, MIOS based unit. I have contacted several people in the particular scenes in question and they have been interested in the project, but it is unlikely they would help until they can get a more solid idea of what is wanted. The possible approach is recorded below and will be continually updated.

## Technical Information

### Overview




The SNES' sound module consist of:

- 1: the sound engine SPC700 8-bit CPU above and DSP unit below, both designed and manufactured by Sony
- 2: 9112 32k SRAM manufactured by Sharp
- 3: D6376 2 channel 16-bit DAC manufactured by NEC
- 4: 9124 32k SRAM manufactured by Hitachi
- 5: 2904 Dual Op-Amp manufactured by JRC
- 6: I/O connector to the mainboard (Notice the staggered connector)

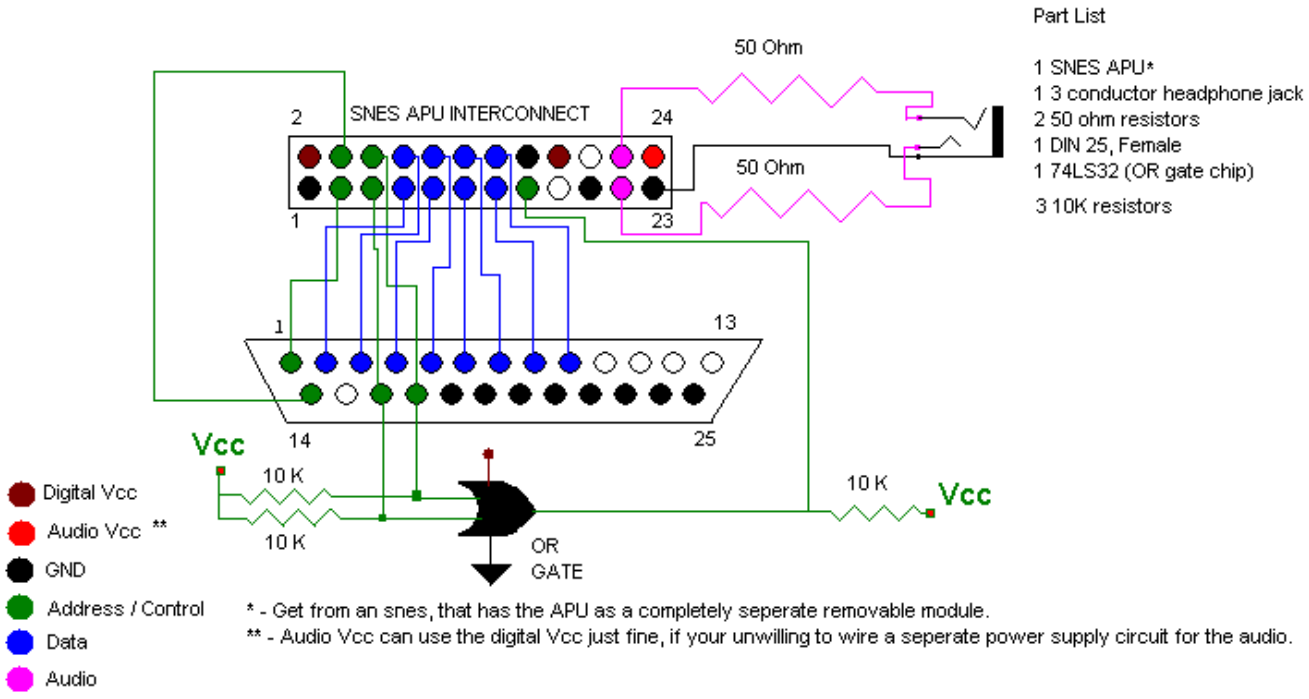
There is a device in the SNES. This device consists of an 8-bit CPU (Sony SPC700) and 64KB of RAM mounted on a circuit board. It's connected to the main board by a series of pins, and communication with it is carried out through one of four ports. Other than that, though, this device is completely separate physically and logically from the console.

The SPC700 runs at a whole 1.024MHz, has six internal registers, and can execute 256 opcodes. Obviously it's not as powerful as the computers of today, but in and of itself it's quite capable of performing many tasks. However, there are no input or display devices attached to it, so what good does it do us?

Well, the SPC700 does have a 16-bit DSP attached. This DSP functions much like the wave table

sound cards today, capable of mixing 8 voices simultaneously at various pitch and volume. Programs can be written for the SPC700 that play music with the DSP just like people write programs for your PC that play music with your sound card.

## Pinout




Using this project as an example one can see just how simple getting an APU connected might be. Though I don't have a SNES I can see from this pinout that it's a rather simple task.

1. PA7
2. PA6
3. PA0
4. PA1
5. /WR
6. /RD
7. D0
8. D1
9. D2
10. D3
11. D4
12. D5
13. D6
14. D7
15. /RESET
16. Not Connected
17. SMPCK
18. Vcc
19. GND
20. Mute
21. Audio Right

22. Audio Left
23. Audio GND
24. Audio Vcc

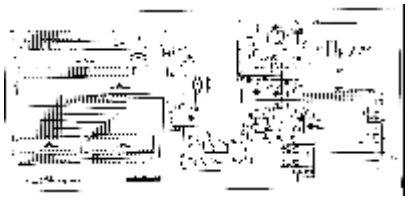
#### Control Address

Pins 1 - 4 control the address. Here is the encoding scheme, with Pin 4 being most significant bit.

1. 2140 = 0010
2. 2141 = 0110
3. 2142 = 1010
4. 2143 = 1110

#### Schematic

I found this schematic, but am having trouble figuring it out. Looks like it will pretty much provide any peripheral circuit components if it can be read (it's fuzzy).


Click for larger view

## Possible Approach

### Intro

It appears that the essential relationship that the APU has with the SNES is one of processor to processor. The SPC700 basically receives a program from the SNES and loads it into its memory each time the cartridge is loaded. The SNES then dynamically tells the APU to play the different sound as the game is played. Music and other information is also sometimes streamed to the APU.

### What this means

With this is mind it appears that the relationship between the core and the APU will be very similar to the SNES situation. Therfore it appears that a program will have to be written for the APU that will allow some sort of passive loading of samples and controlling of the synthesizer. I will have to facilitate smooth communications between the two units using the data lines of the APU.

## To Do

1. Find a SNES

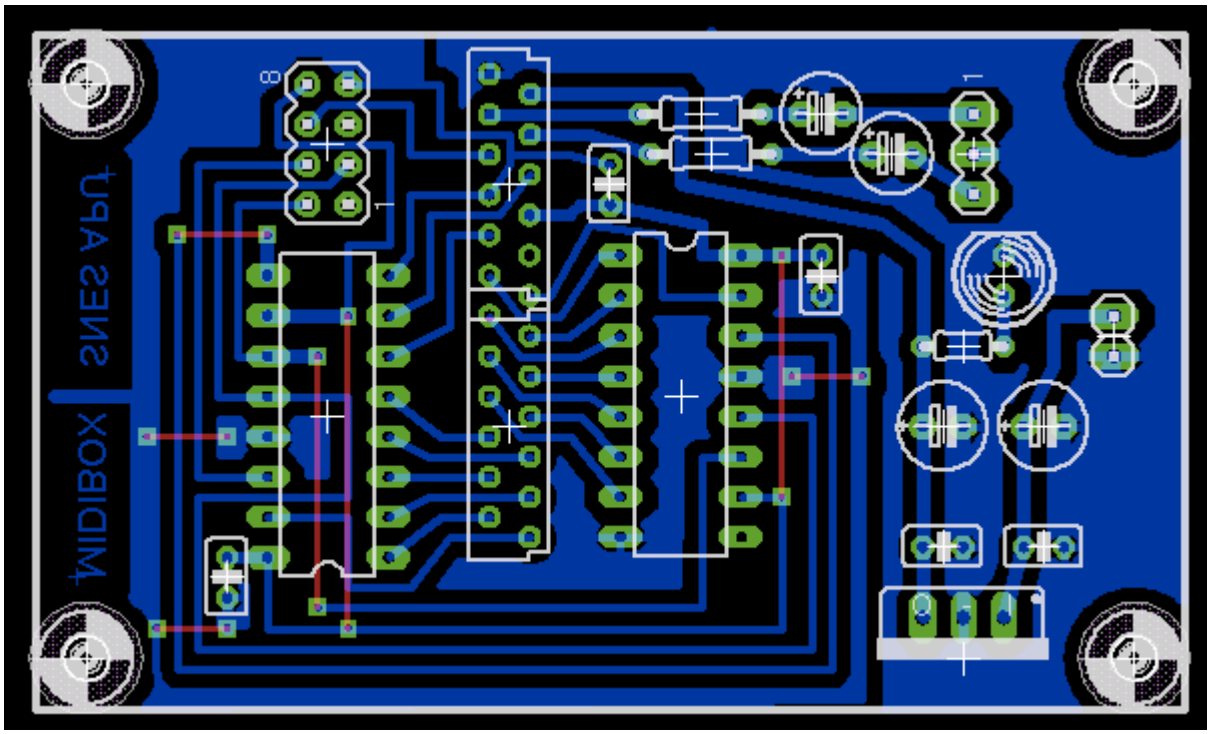
2. Board for the APU to connect to a Core
    1. Consult actual SNES components
        1. What's the clock frequency?
            1. UPDATE: Might be 24.576 MHz
        2. Should there be an external oscillator, and what kind?
        3. What with the WD & WR lines and how to use the AND OR? Present design just uses parallel control
        4. Power consumption?
    2. Consult past MIDIBOX projects
    3. Ask someone who knows about MIDIBOX
        1. Memory required?
        2. How would it connect to the core (there are two unused ports in the J2)?
3. Software for MIOS
4. Software for APU
    1. Find a emulation/homebrew scene person who can help

# Present Status

Just started this page. Have hardly begun even entering or searching through the information I have. I even have a preliminary board design for the module but there is a lot to be desired at this point.

### Preliminary Schematic

I don't really know how I should be setting this thing but, I'm mostly going based on the SID module (Shift register but a lot of the info is cloudy. Will have to figure out what I can do for the other options for the chip (not just data lines).



# Links

## Information

- Good SPC overview http://snesmusic.org/files/spc700.html
- SNES Developer's Lobby http://filebox.vt.edu/users/rogrubb3/SNESDev/
- More hardware information http://emureview.ztnet.com/developerscorner/
- Pinout http://www.raphnet.net/electronique/snes_apu/APU_pinout.txt
- APU on a PC http://www.raphnet.net/electronique/snes_apu/snes_apu_en.php
- SPC700 (japanese) http://www.sony.co.jp/Products/SC-HP/tec/micom/spc700/index.html
- WikiBooks: Writing SPC programs http://en.wikibooks.org/wiki/Super_NES_Programming/Loading_SPC700_programs
- Zophar's Domain SPC700 Info http://www.zophar.net/tech/snes.html
- SPC Programming http://www.emulatronia.com/doctec/consolas/snes/spc.htm

## Forum Posts

- Forum posting by me http://board.zsnes.com/phpBB2/viewtopic.php?p=134847#134847
- MIDIBOX SNES-APU? http://www.midibox.org/forum/index.php?topic=4975.0
- MIDIBOX NES? http://www.midibox.org/forum/index.php?topic=4555.0
- OFFTOPIC http://www.midibox.org/forum/index.php?topic=6473.0
- SPC File Format http://board.zsnes.com/phpBB2/viewtopic.php?t=7530&highlight=spc700
- SPC Tracer http://board.zsnes.com/phpBB2/viewtopic.php?t=7173&highlight=spc700
- SNES RPG http://linksadventure.no-ip.org/phpbb2/viewtopic.php?t=72
- RomHacking: Making SPC Music http://www.romhacking.net/forum/index.php/topic,1959.0.html
- RomHacking: SPC Creation http://www.romhacking.net/forum/index.php/topic,636.0.html

## Other SPC Projects

- A SPC tracker project http://ekid.nintendev.com/xms/

From:
https://www.midibox.org/dokuwiki/ - **MIDIbox**

Permanent link:
**https://www.midibox.org/dokuwiki/doku.php?id=midibox_snes_apu**

Last update: **2011/09/17 17:00**