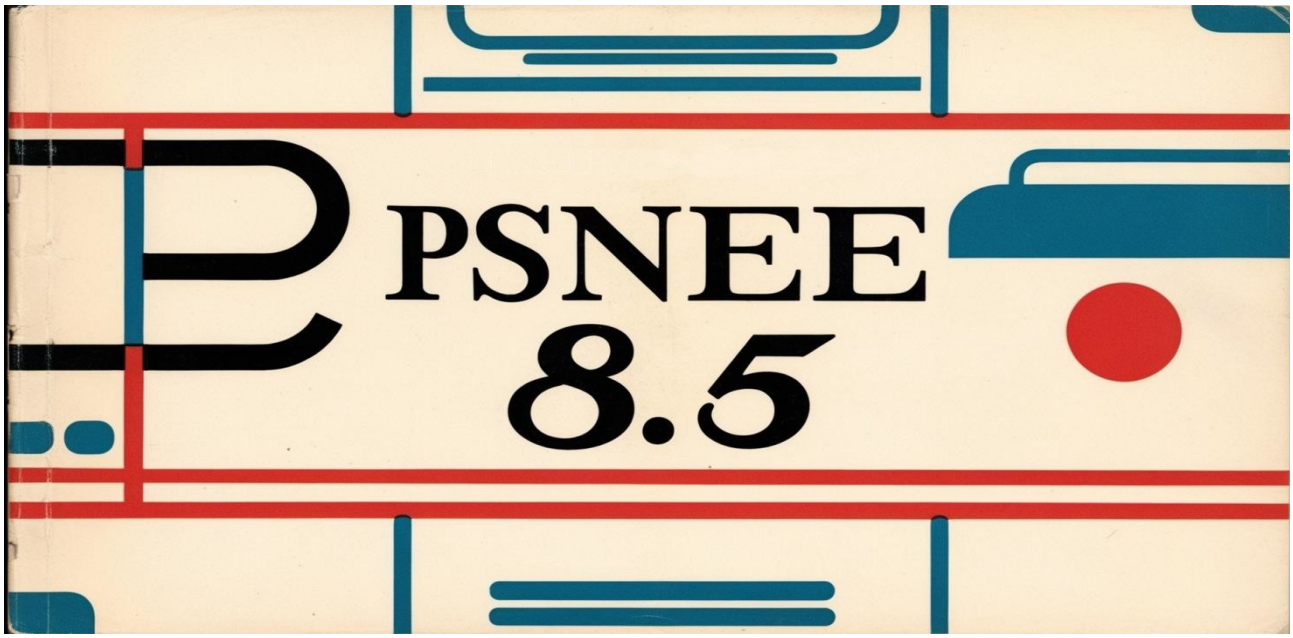
PsNee modchip installation guide



Many thanks to William Quade, who allowed me to use his work as the foundation for this guide https://quade.co/.

The PsNee modchip is a newer chip that has been developed over the last few years and is still under active development. PsNee modchips should work with all versions of the PlayStation 1. They can be made using many AVR processors, including,

- ATmega328/168 (Arduino Uno, Arduino Mini/Mini Pro)
- ATmega32u4 (Arduino Leonardo, Arduino Micro/Micro Pro)
- ATtiny25/45/85
- LGT8F328 (copy of ATmega328)

This guide covers everything you need to know about choosing, programming, and installing a PsNee modchip in your PlayStation.

Guide structure

This guide is structured into several pages. It's organized like this primarily so that comments can be more organized and useful for readers.

This page covers the PsNee modchip. It includes general information about the chip, and then links off to installation guides for each console model.

If you want more information about PlayStation 1 modchips in general check out this guide.

PsNee modchip drawbacks

The biggest problem with PsNee chips is that they aren't very well documented, in terms installation diagrams for every board revision.

This guide, will help alleviate thes issue by providing installation diagrams.

PsNee modchip advantages

The greatest strength of the PsNee modchip is the fact that it's much more modern than PIC based modchips. Using AVR chips means that programming a PsNee chip is much easier, and many people may already have something like an Arduino which can be used as a PsNee modchip.

Using more powerful processors allows the modchip to work with American, European, and Japanese PlayStation 1's with a single chip, no need for a version for each region. It can even patch the BIOS Note that this BIOS patch dont works with ATtiny chips.

PsNee modchip source code

If you'd like to program your own PsNee modchips you'll need the source code. It supports the following microcontrollers:These chips fall into two categories, ATmega based chips, and ATtiny based chips. ATmega chips cost more, but allow for BIOS patching. ATtiny chips cost less. ATmega chips are also considerably larger in size, making them more difficult to fit into a system when compared to the ATtiny chips.

The source code is available on the PsNee Github page. It can be compiled very easily using the Arduino IDE.

Below is a list of all PlayStation 1 motherboard versions, along with the console model numbers associated with them. You can get a good idea of what board you have by looking at the model number underneath your system (something like SCPH-7501).

To actually know what board you have you'll most likely have to open up your console and look for the board version printed somewhere on the board (something like PU-22 or PM-41).

Once you know what board you have you can click on the board model and you'll be sent to a page with the installation diagram.

- PU-7
  - All SCPH-1000's
  - Some early SCPH-1001, and SCPH-1002's
  - Some early SCPH-3000's
  - Some early SCPH 3500's
- PU-8
  - Most SCPH-1001, and SCPH-1002's
  - Most SCPH-3000's
  - Most SCPH-3500's
  - All SCPH-5000's
- PU-16
  - All SCPH-5903's
- PU-18
  - All SCPH-5001's
  - All SCPH-5500, SCPH-5501, SCPH-5502, and SCPH-5503's
  - All SCPH-5552's
  - Some early SCPH-7000, SCPH-7001, SCPH-7002, and SCPH-7003's
  - Some early SCPH-7501's
- PU-20
  - Most SCPH-7000, SCPH-7001, SCPH-7002, and SCPH-7003's
- PU-22
  - All SCPH-7500's
  - Most SCPH-7501's
  - All SCPH-7502, and SCPH-7503's
  - Some early SCPH-9000, SCPH-9001, SCPH-9002, and SCPH-9003's
- PU-23
  - Most SCPH-9000, SCPH-9001, SCPH-9002, and SCPH-9003's
- PM-41
  - Earlier SCPH-100, SCPH-101, SCPH-102, and SCPH-103's
- PM-41 (2)
  - Later SCPH-100, SCPH-101, SCPH-102, and SCPH-103's

Quick summary of compilation and installation.

- Choosing your compilation options.
- Compilation.
- Configure fuses.
  - For BIOS patch, H: DF, L: EE, E: FF
  - Without BIOS patch, H: DF, L: FF, E: FF
- Injecting the code into the target via ISP.

- Soldering mode according to the diagrams.

Personal recipe to make a good PSNee.

List of ingredients:

Two Arduino cards (for example a nano, and a Nano Pro).

A soldering iron (the tTS_100 found in all Chinese stores is a good compromise)

A little soldering wire

And thin cable (awg26-30)

Part 1:

Install the Arduino IDE

https://docs.arduino.cc/software/ide-v2/tutorials/getting-started/ide-v2-downloading-and-installing/

Part 2:

Check that everything works well by injecting for example the blink sketch from the IDE

File -> example -> 01. Basic -> blink

Tools -> card -> arduino AVR board -> your board intended to become the ISP gateway

Tools -> port -> the port of your card

Tools -> programmer -> arduinoISP

If as heard your card flashes you can go to the next step

Part 3:

Use this card to transform it into an ISP gateway

https://docs.arduino.cc/built-in-examples/arduino-isp/ArduinoISP/

Plug it in to become the PSNee, do the test again with the blink sketch by changing the program.

Tools -> programmer -> arduino as ISP

If you succeeded, the serious stuff begins.

Part 4:

Install the PSNeecore file in the arduino15/pakages/ directory

https://support.arduino.cc/hc/en-us/articles/360018448279-Open-the-Arduino15-folder

Part 5:

Selecting options in the sketch.

Open the sketch to Arduino, select in the PSNee.ino the MUC to become the PSNee, choose the target SCPH-xxxx.

Part 6:

In Arduino IDE select MUC buildchain

Tools -> board -> PSNee -> your MUC

Part 7:

Configure fuses (not available for ATtiny25/45/85)

Tools -> fuses -> [BIOS patch] if your board needs it, if not [NO BIOS patch]

Part 8:

Compile and inject the program.

Sketch -> verify/compile

Sketch -> upload using a programmer

Normally your PSNee is ready to be installed in your dear console.

Part 9:

Install your PSNee in your console according to the schematics available in this guide.

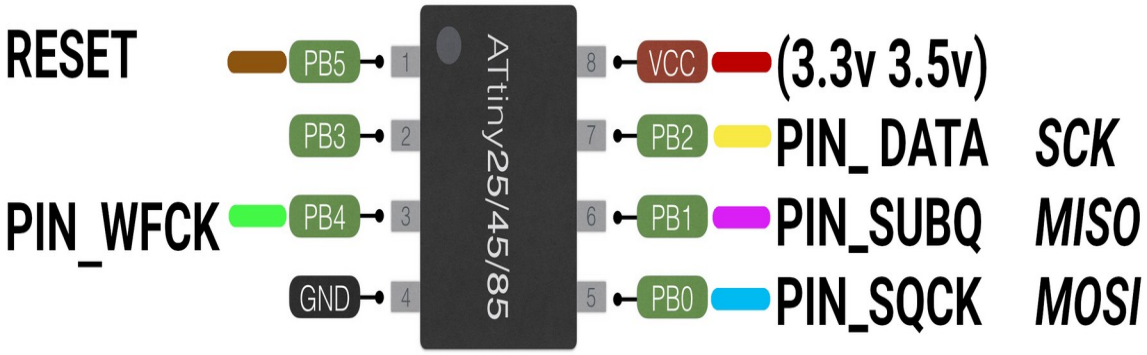
If your console does not require a bios patch, ignore the AX AY DX RESET pins.

If your console requires a bios patch, for the AX AY DX pins follow the BIOS Pin diagram.

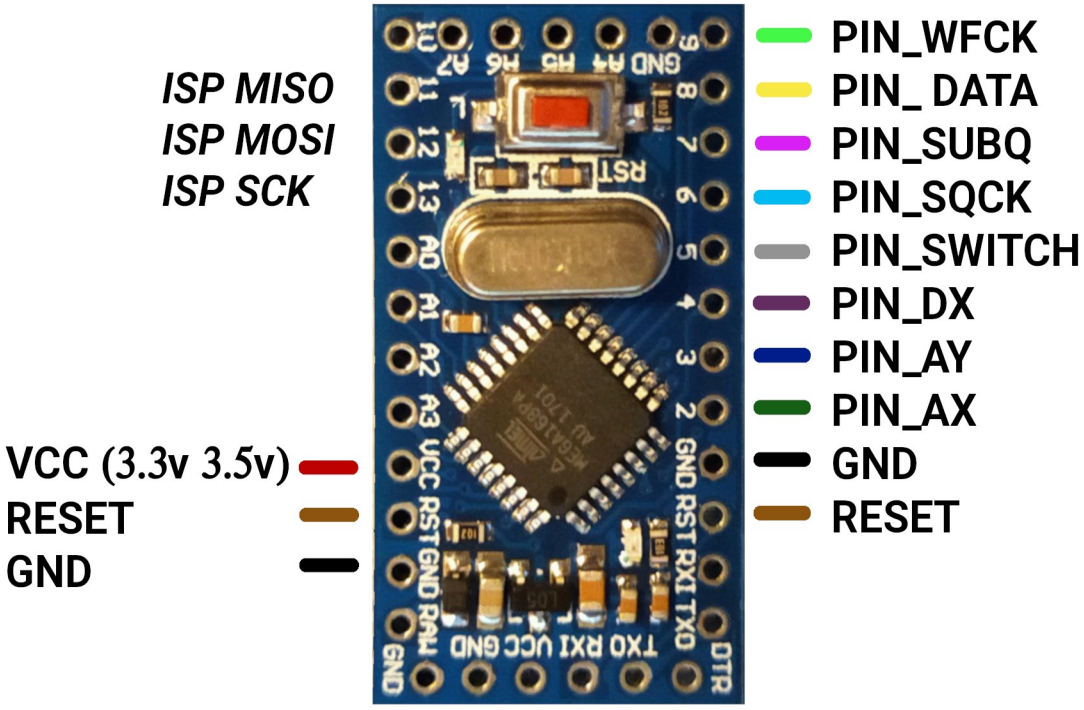
Now you can enjoy your delicious pancakes hot off the burner on your PS1;P

PSNee installation diagrams & pinout

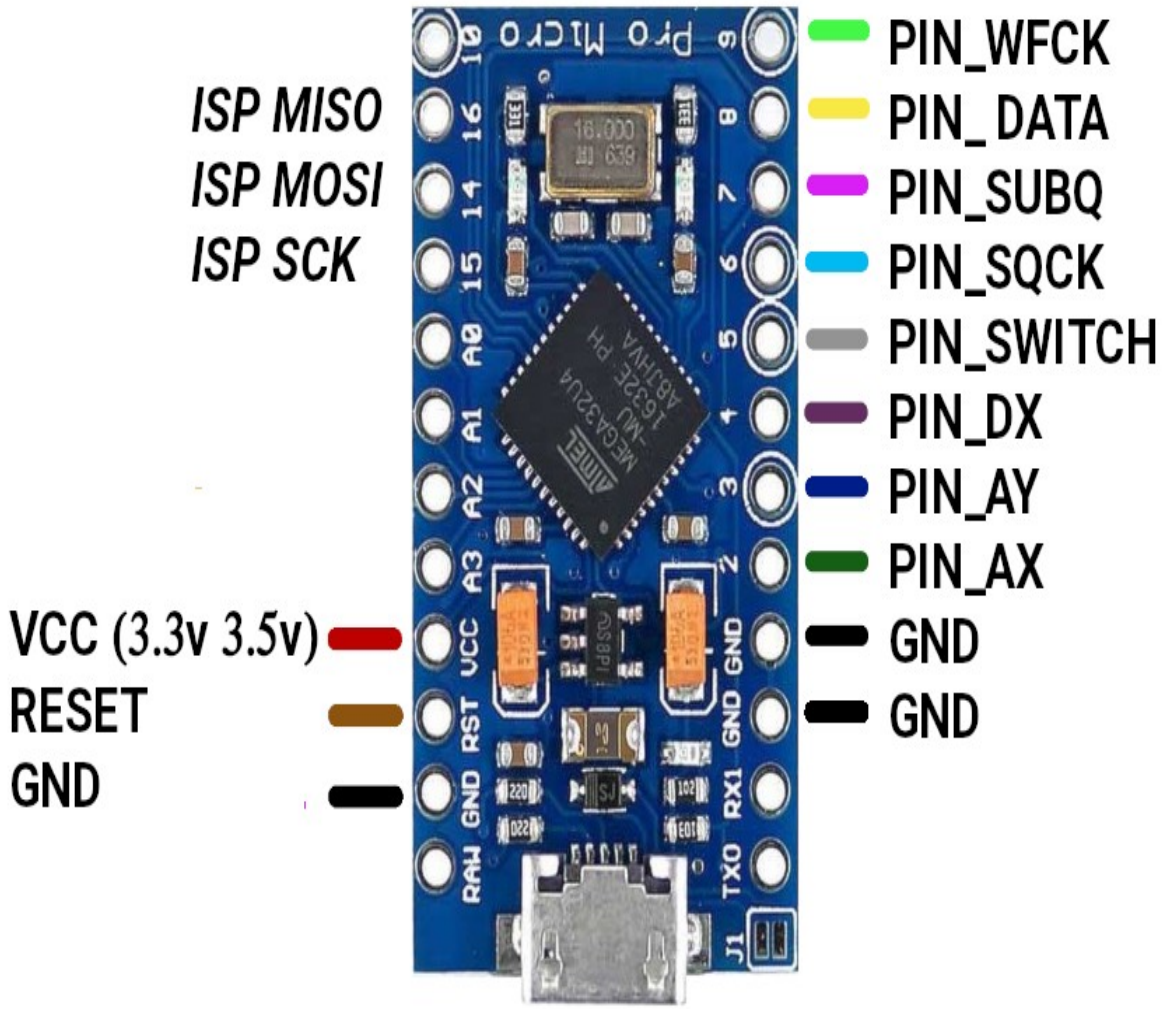
ATtiny25/45/85 pinout



Arduino Pro Mini Pinout
ATmega328_168



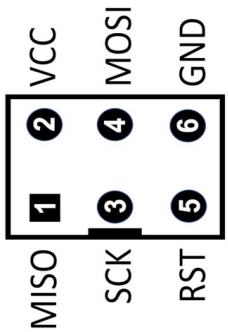
Arduino Pro Micro pinout
ATmega32U4_16U4



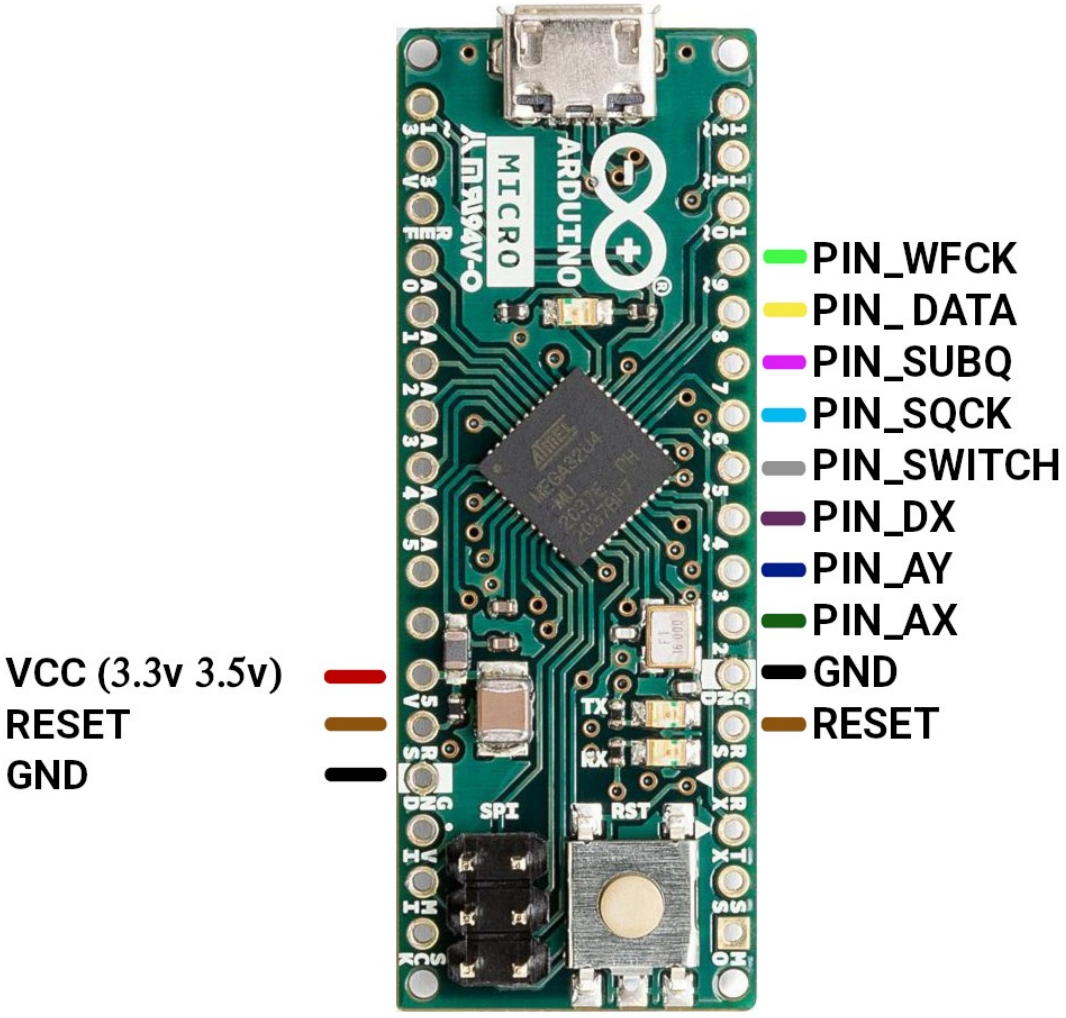
Arduino Nano Pinout
ATmega328_168

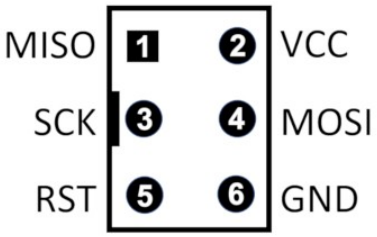
VCC (3.3v 3.5v)
RESET
GND

PIN_WFCK
PIN_ DATA
PIN_SUBQ
PIN_SQCK
PIN_SWITCH
PIN_DX
PIN_AY
PIN_AX
GND
RESET

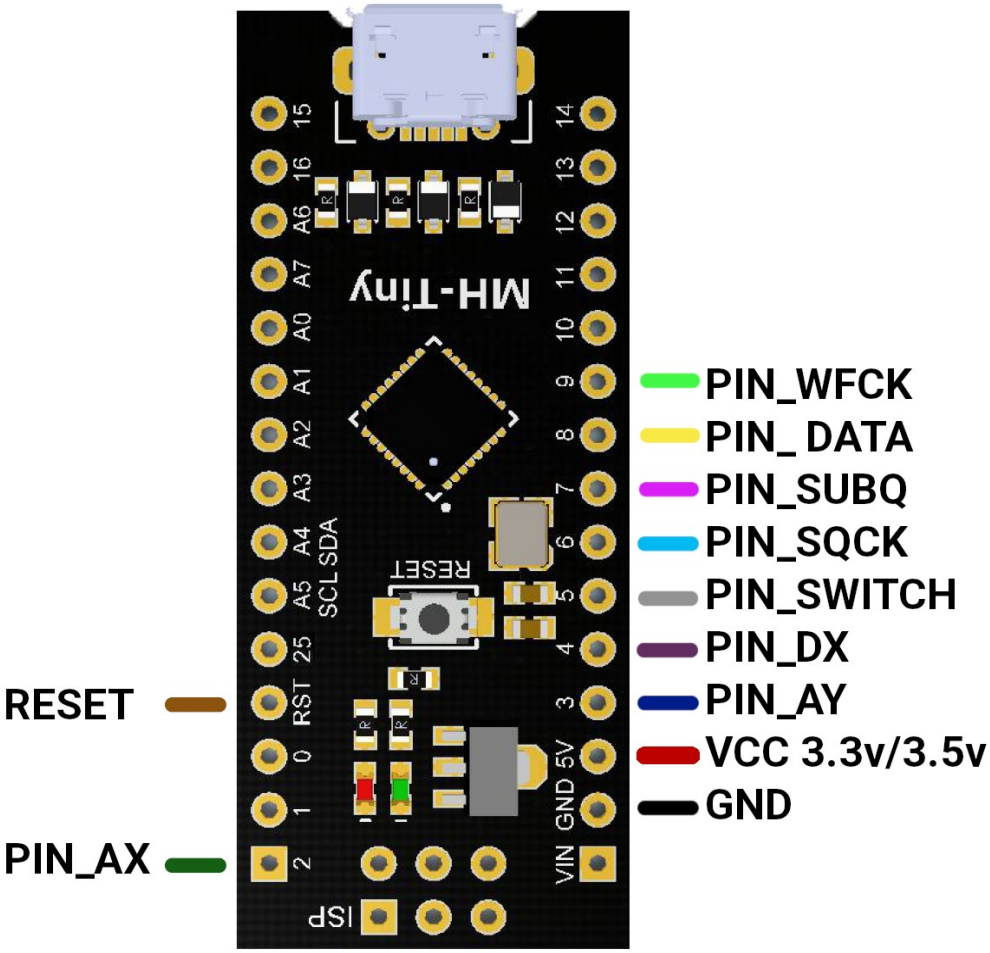


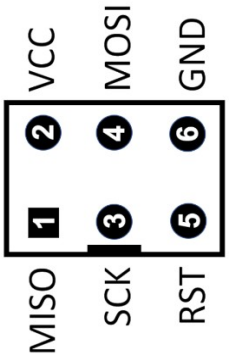
Arduino Micro pinout
ATmega32U4_16U4





ATTINY88 NANO PINOUT
ATtiny88



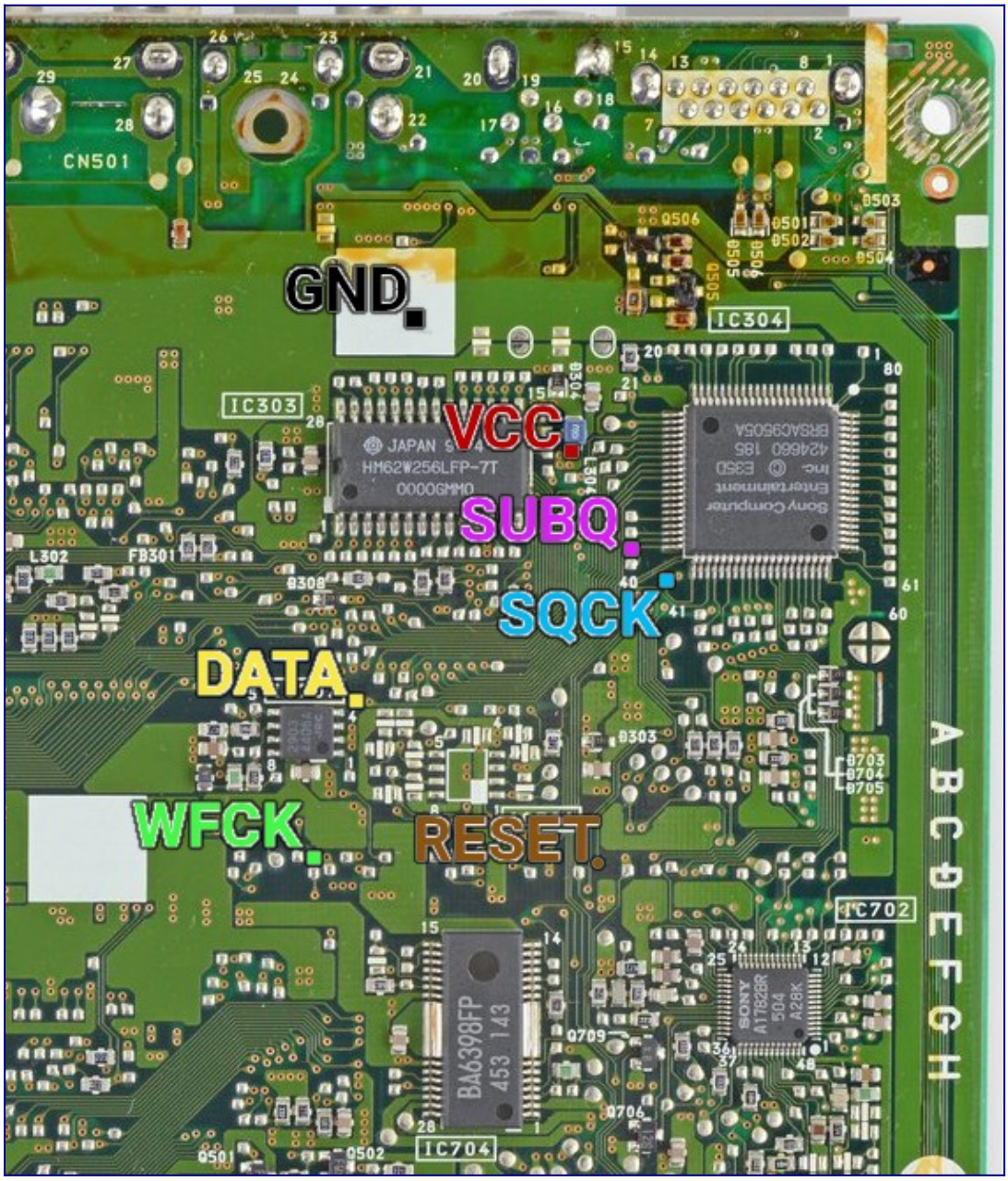


PU-7 PsNee modchip installation diagram

The PU-7 was the very first PlayStation 1 board that was released. It is found on all Japanese SCPH-1000's, along with some SCPH-1001's, SCPH-1002's, SCPH-3000's, and SCPH-3500's.

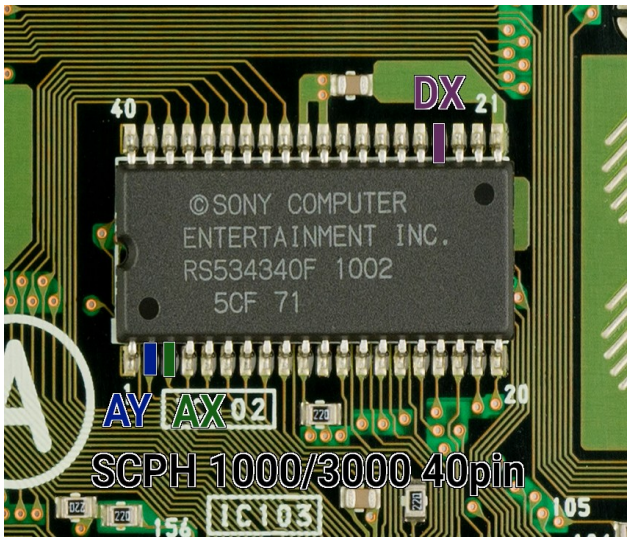
This board is unique because it has the pins for S-Video video output along with the RCA video output ports. The S-Video port was only available on the Japanese SCPH-1000, but the pins are still on other systems with the PU-7 board.

PU-7 PsNee modchip installation diagram



Above is the diagram for the PU-7.

On the side you have the diagram for the BIOS of the pu-7.



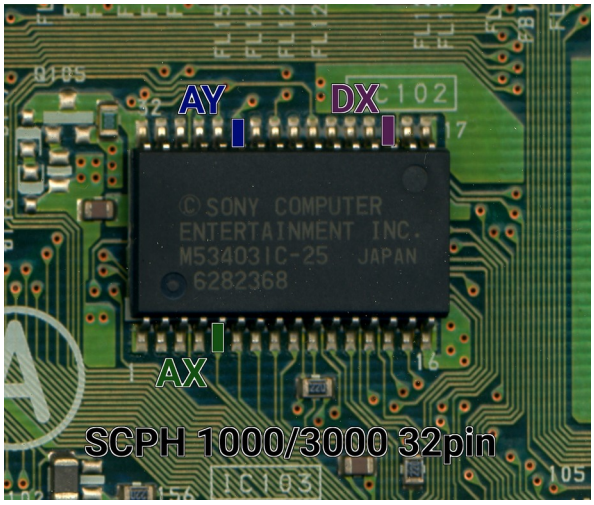
About the points

For models SCPH-xxx1, SCPH-xxx2, all the points are on the back, and you can ignore the RESET. On the other hand, the model and SCPH_xxx0, you will have to use the RESET point, and the BIOS points.

Installation tips

Here are some tips I have for you when you are soldering your chip into the PU-7.

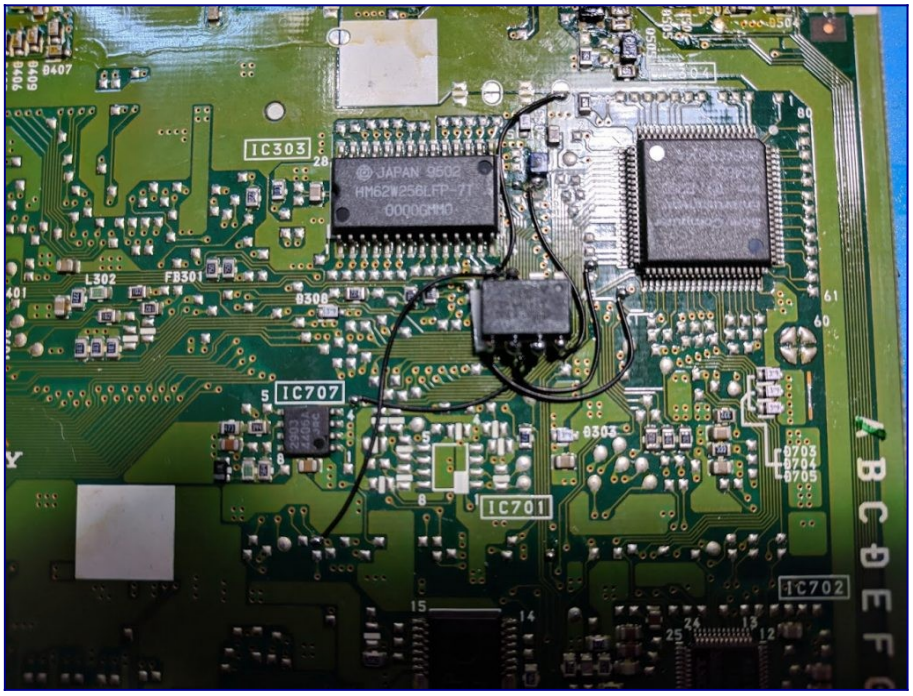
- **Cut your wires to be as short and direct as possible.**
- You don't need to connect pins one and two of the ATtinyX5 chip. Just desolder the wire.
- Use a multimeter to probe around for alternative VCC and GND points closer to where you position your chip for a cleaner installation.
- Try to position your chip towards the middle of where all the wires need to go, to minimize wire length.



Example installations

This section has photos of some successful installations which you can use to get a better understanding of how everything is wired and positioned.

This is an installation I did on an SCPH-1000:

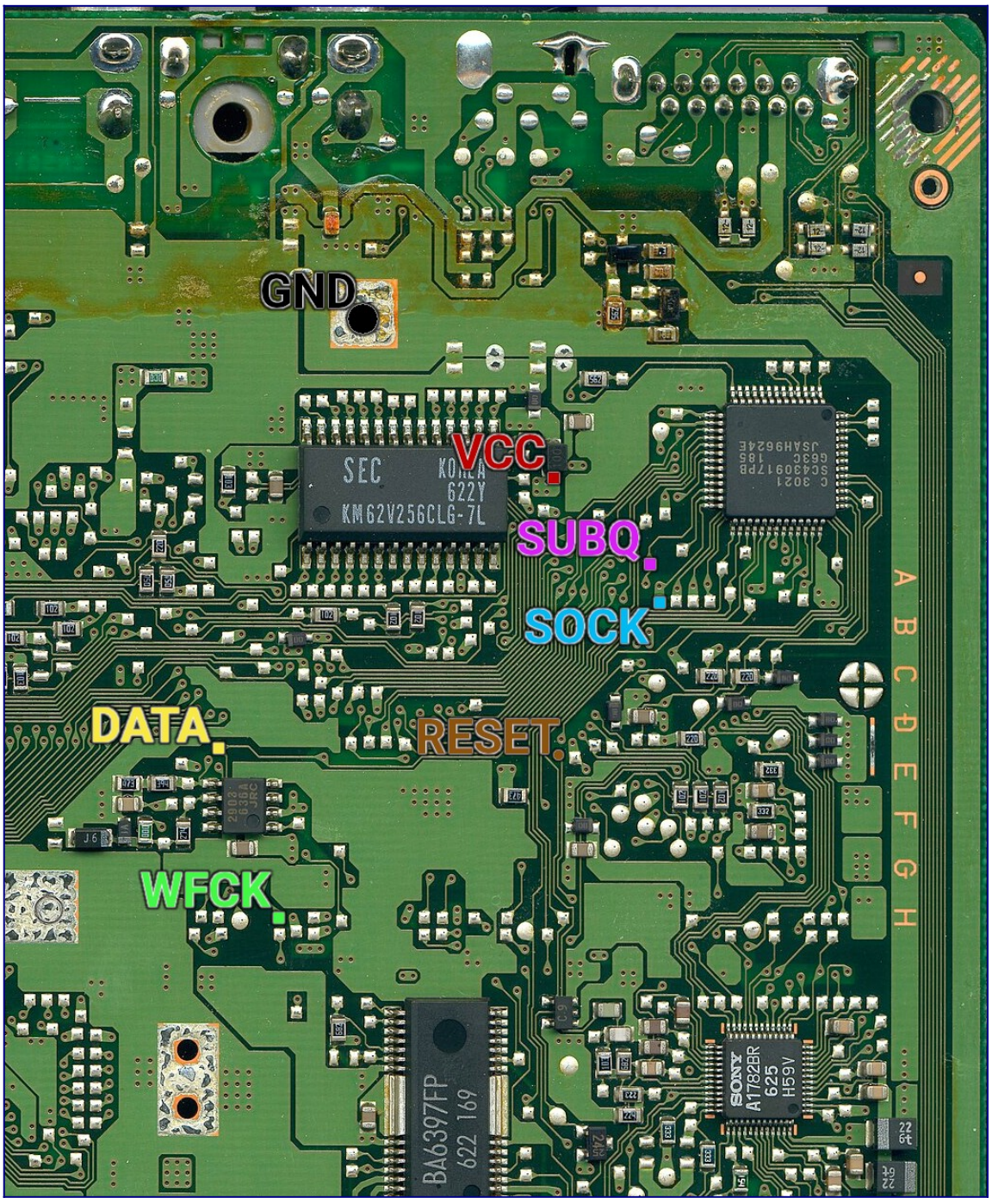


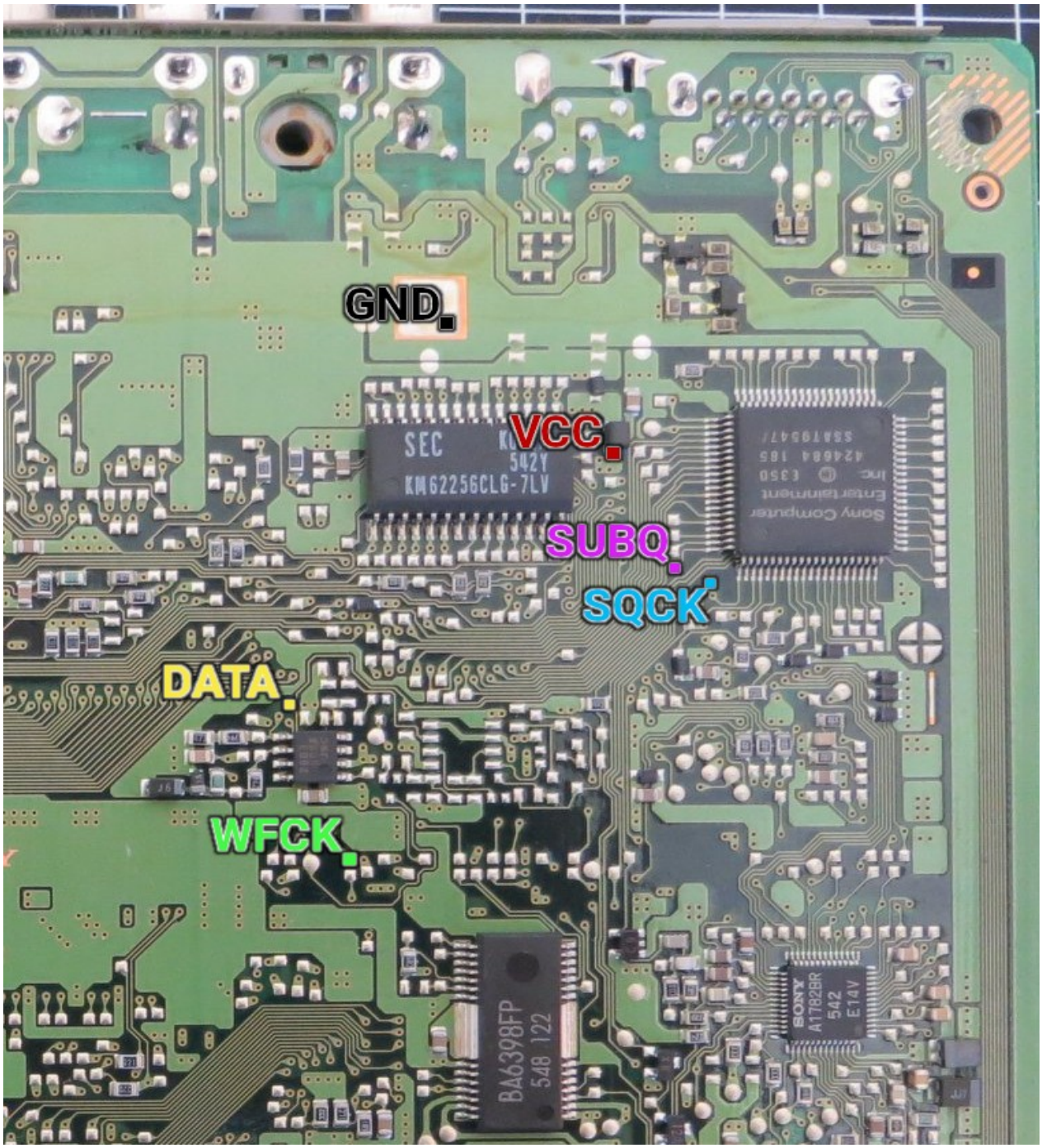
PU-8 PsNee modchip installation diagram

This was one of the earlier PlayStation 1 board designs that included RCA composite video output ports on the back. Most SCPH-1001’s here in the United States had this board inside.

Additionally many Japanese systems had this board including many SCPH-3000, SCPH-3500, and SCPH-5000’s.

PU-8 PsNee modchip installation diagram



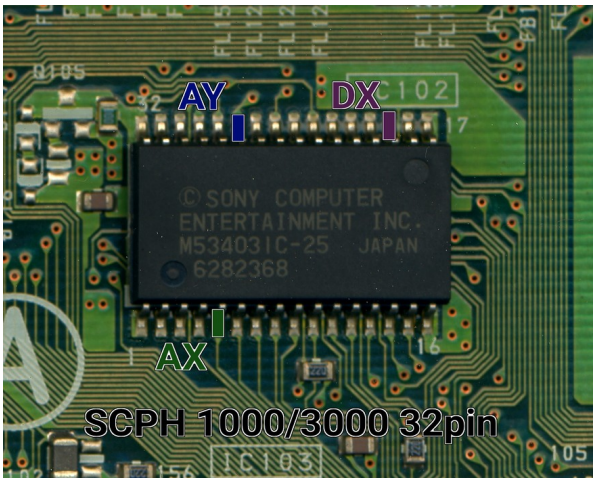


Above are two installation diagrams for two different versions of the PU-8. The easiest way to know which you have is to look at the size of the Mechacon controller chip (next to SUBQ and SCLK/SQCK). If it’s large, use the bottom diagram, if it’s small, use the top diagram.

On the side the points for the BIOS of the pu-8

About the points

For models SCPH-xxx1, SCPH-xxx2, all the points are on the back, and you can ignore the RESET. On the other hand, the model and SCPH_xxx0, you will have to use the RESET point, and the BIOS points.



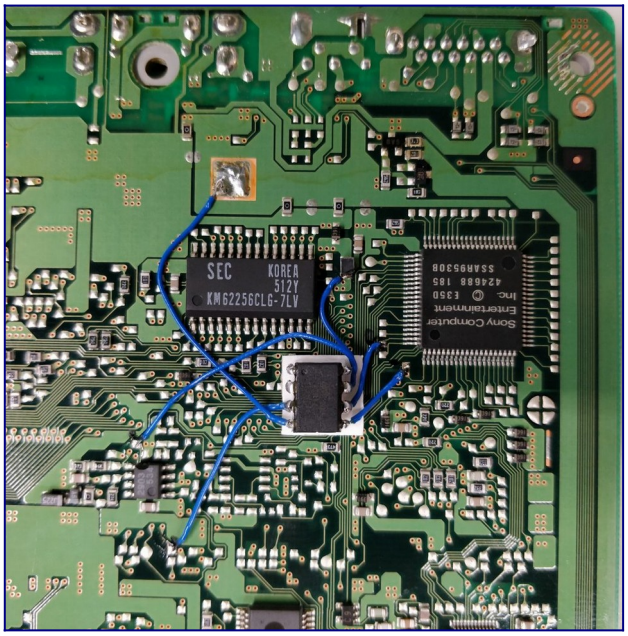
Installation tips

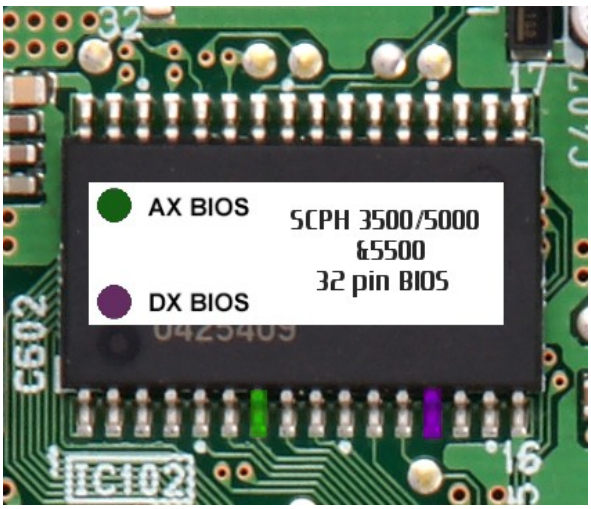
Here are some tips I have for you when you are soldering your chip into the PU-8.

- **Cut your wires to be as short and direct as possible.**
- You don’t need to connect pins one and two of the ATtinyX5 chip. Just desolder the wire.
- Use a multimeter to probe around for alternative VCC and GND points closer to where you position your chip for a cleaner installation.
- Try to position your chip towards the middle of where all the wires need to go, to minimize wire length.

Example installations

This section has photos of some successful installations which you can use to get a better understanding of how everything is wired and positioned.





PU-16 PsNee modchip installation diagram

According to Wikipedia the PU-16 was only found in the SCPH-5903. This is a very unique board in that it was only released for the Asian market, and was only in a single model, and it was colored white instead of grey.

The SCPH-5903 is the only PS1 model that can play video CD movies. The system also has the RCA video plugs like the original SCPH-100x models did.

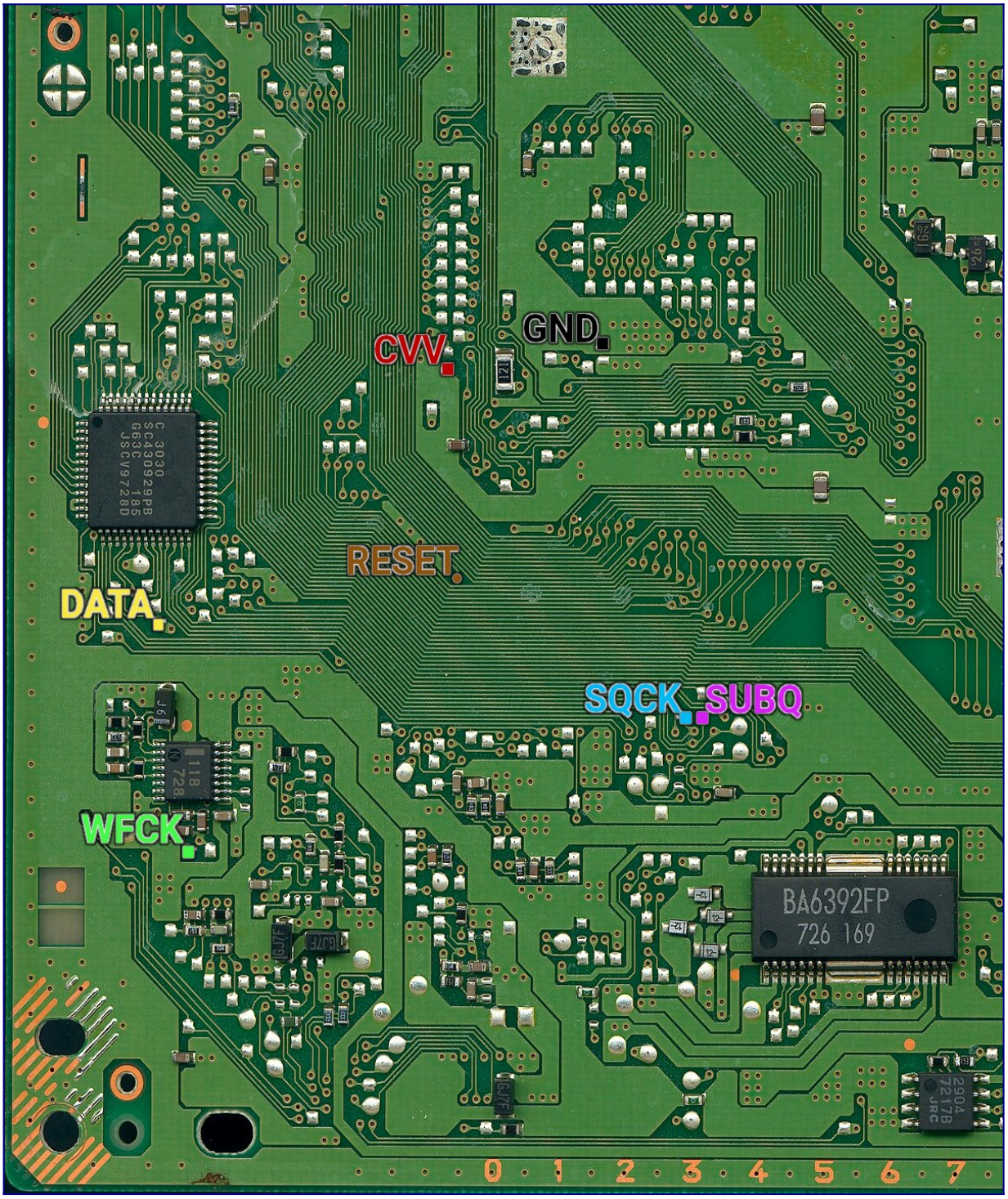
Apparently there were only around 10,000 of these made, and they were primarily in Hong Kong. So they’re fairly rare. Because of this I can’t find any PsNee installation diagrams for the system.

Based on what I can find online the board most resembles the SCPH-1000. If you have one of these I’d look at PsNee diagrams for the SCPH-1000 and other earlier boards and try to figure out where things go through trial and error. In other words take a look at the PU-7 and PU-8 diagrams.

PU-18 PsNee modchip installation diagram

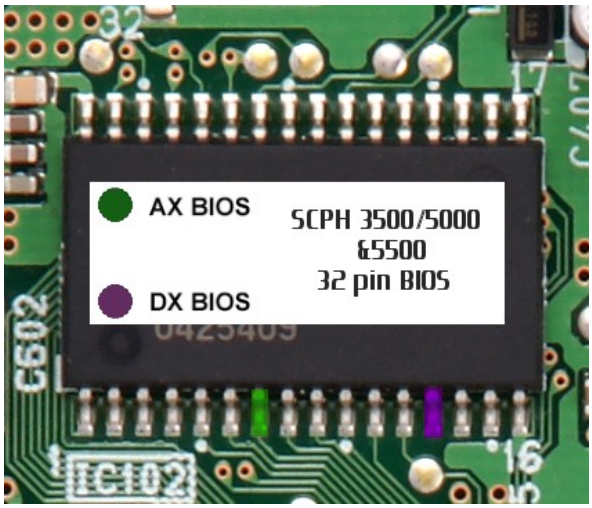
This board was used with a wide range of consoles from the SCPH-5000’s, and SCPH-5500’s, to even some SCPH-7000’s and SCPH-7500’s.

PU-18 PsNee modchip installation diagram



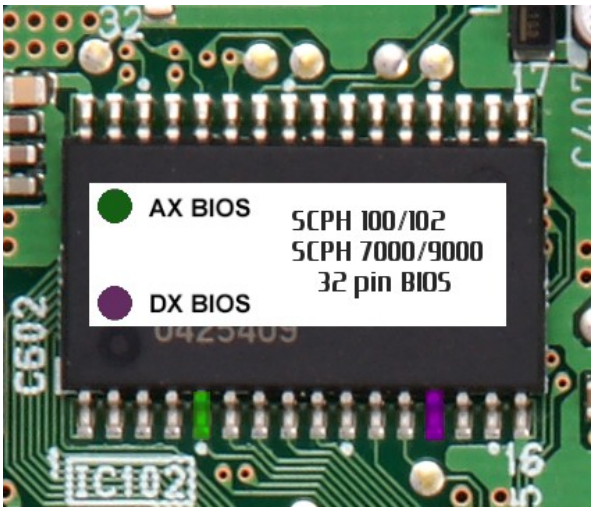
Above is the diagram for the PU-18.

On the side the points for the BIOS of the PU-18



About the points

For models SCPH-xxx1, SCPH-xxx2, all the points are on the back, and you can ignore the RESET. On the other hand, the model and SCPH_xxx0, you will have to use the RESET point, and the BIOS points.All of the points are fairly easy to solder to.



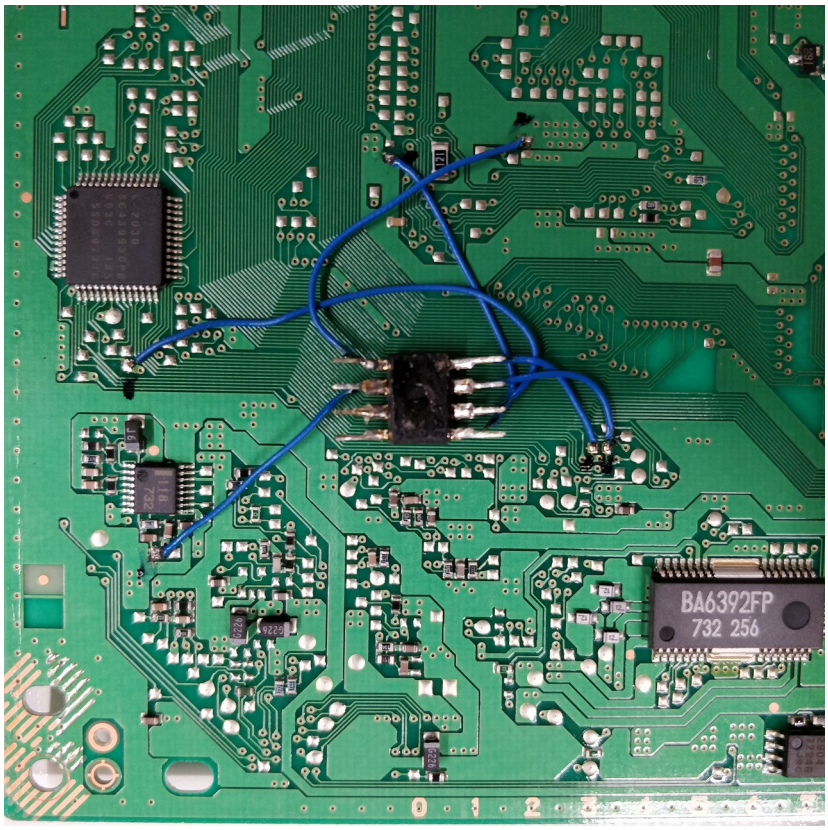
Installation tips

Here are some tips I have for you when you are soldering your chip into the PU-18.

- **Cut your wires to be as short and direct as possible.**
- You don’t need to connect pins one and two of the ATtinyX5 chip. Just desolder the wire.
- A good place to put the chip is underneath where the legend is, towards the middle of everything.
- Use a multimeter to probe around for alternative VCC and GND points closer to where you position your chip for a cleaner installation.

Example installations

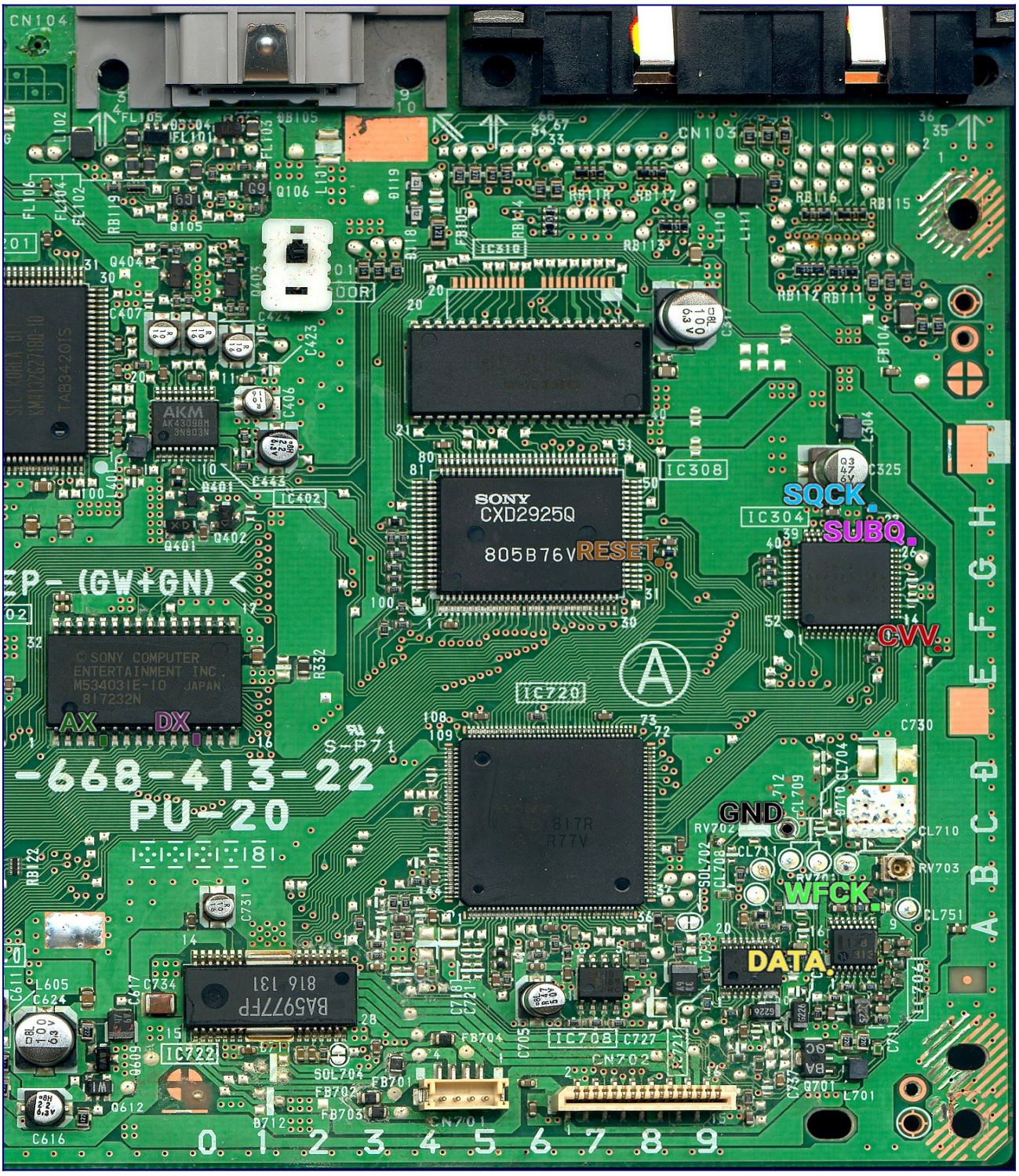
This section displays example installations to help you get a better understanding what a successful installation looks like.



PU-20 PsNee modchip installation diagram

This board was used exclusively with the SCPH-7000 series of consoles.

PU-20 PsNee modchip installation diagram



Above is the installation diagram for the PU-20. Just match each colored pin label in the diagram with any matching colored point on the board.

About the points

For models SCPH-xxx1, SCPH-xxx2, you can ignore the RESET. On the other hand, the model and SCPH_xxx0, you will have to use the RESET point, and the BIOS points.All of the points are fairly easy to solder to

Installation tips

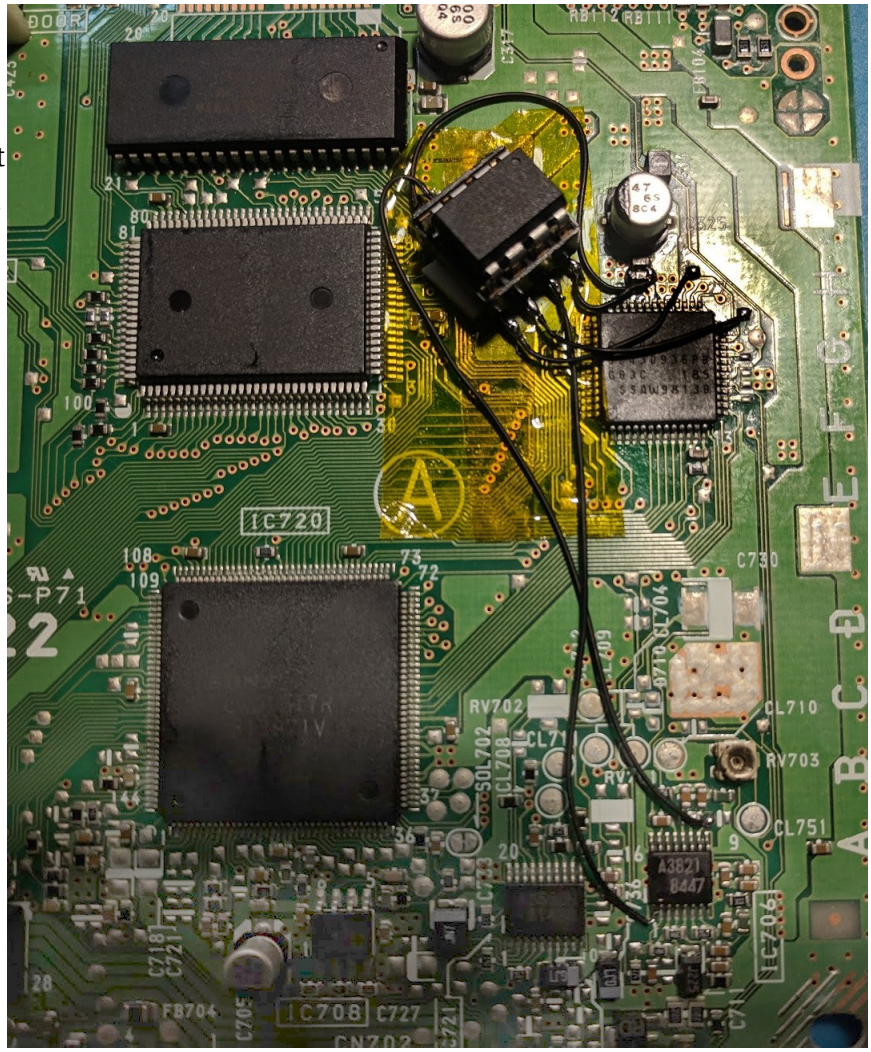
Here are some tips I have for you when you are soldering your chip into the PU-20.

- **Cut your wires to be as short and direct as possible.**
- You don’t need to connect pins one and two of the ATtinyX5 chip. Just desolder the wire.
- The DATA pin can be trickier to solder, make sure you don’t use too much solder or you risk bridging pins on the chip.

Example installations

This section has photos of successful installations which you can use to get a better understanding of how everything is wired and positioned.

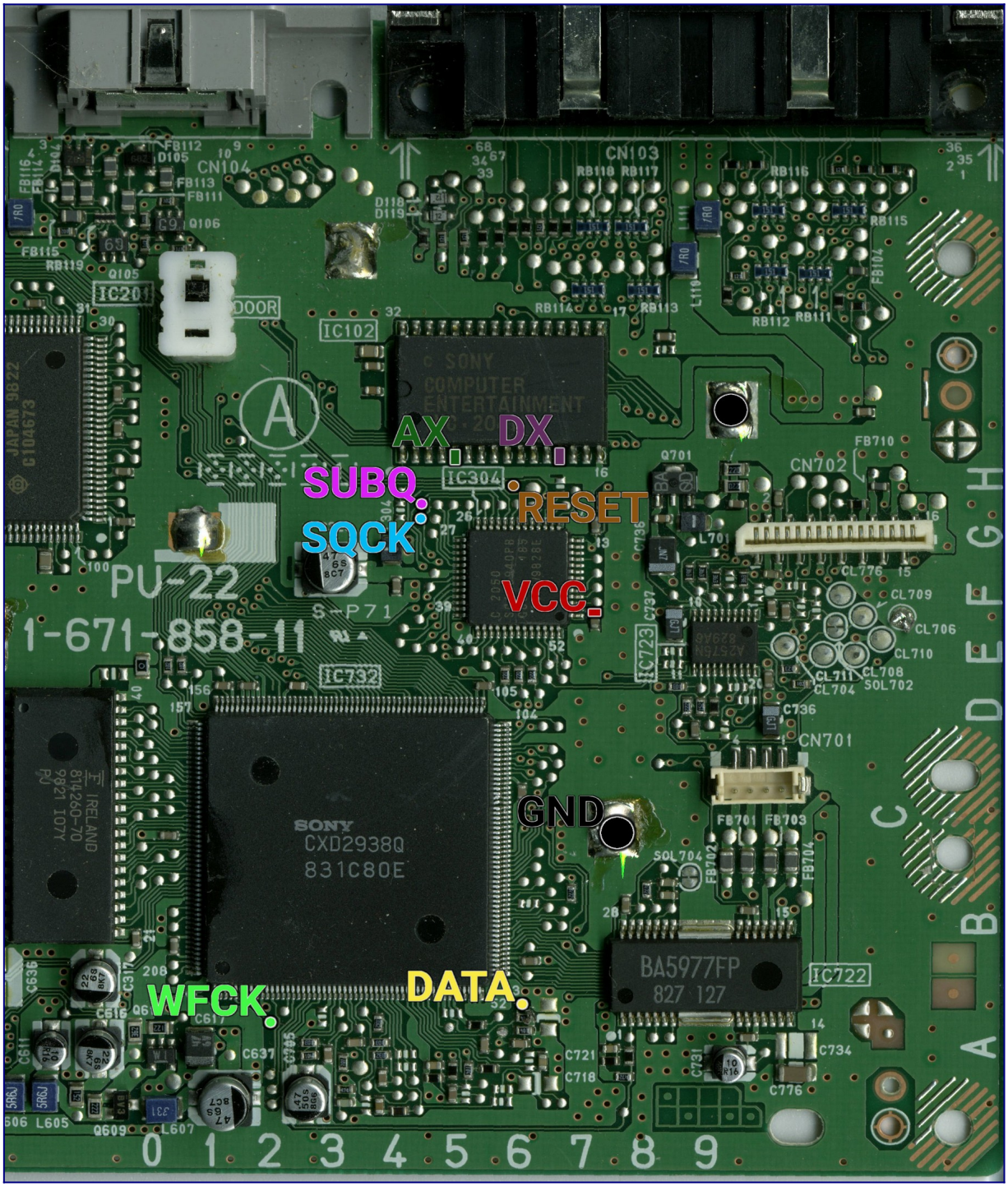
This is an example of an installation I did in a PU-20 using a socket. Using the socket allows me to test chips, and test new PsNee code. Note that the chip is a little too tall and prevents the top metal shield from going on all the way.



PU-22 PsNee modchip installation diagram

This is my favorite version of the PlayStation 1. It’s in newer systems, so the lasers are more likely to still be in good condition, but the system still has both the serial and parallel ports on the back. This board was used primarily with the SCPH-7500 series of systems.

PU-22 PsNee modchip installation diagram



Above is the diagram for the PU-22.

About the points

For models SCPH-xxx1, SCPH-xxx2, you can ignore the RESET. On the other hand, the model and SCPH_xxx0, you will have to use the RESET point, and the BIOS points.

All of the points are fairly easy to solder toThis diagram is fairly straightforward. Be careful when soldering to the capacitor that connects to VCC and GND. In particular, the GND point can easily be bridged to nearby pins/pads.

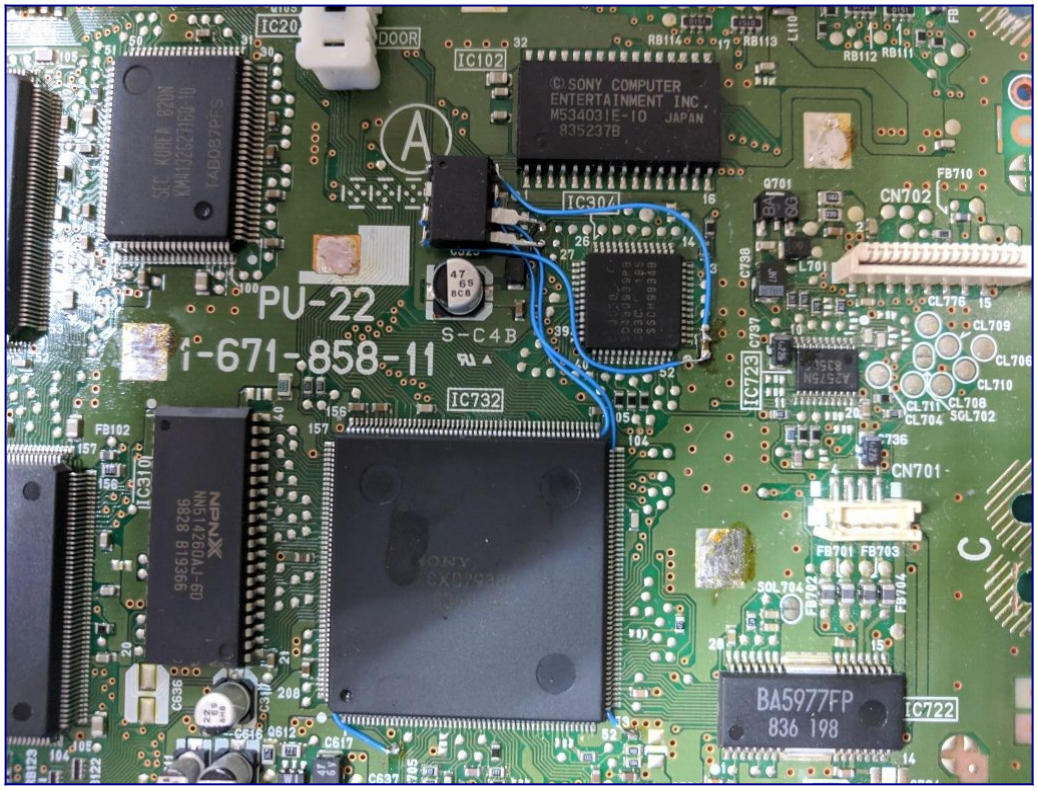
Installation tips

Here are some tips I have for you when you are soldering your chip into the PU-22.

- **Cut your wires to be as short and direct as possible.**
- You don’t need to connect pins one and two of the ATtinyX5 chip. Just desolder the wire.
- Use a multimeter to probe around for alternative VCC and GND points closer to where you position your chip for a cleaner installation.
- Placing the chip on top of the chip in between SUBQ, SQCK, GND, and VCC is a good spot.
- Watch out for solder splatter when you are desoldering the metal shield on top of the board. It isn’t necessary to put it back on, but I always do.

Example installations

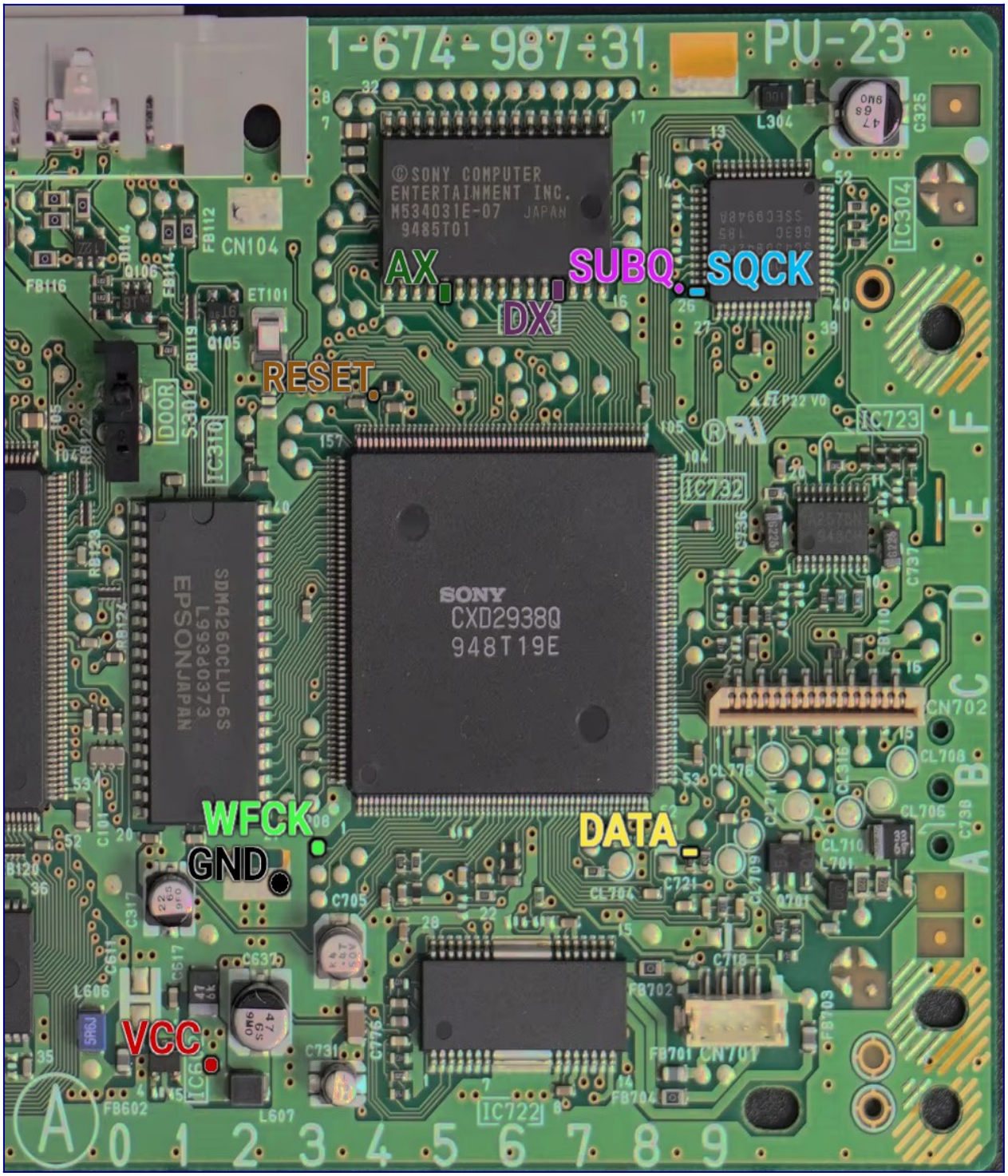
This section has photos of some successful installations which you can use to get a better understanding of how everything is wired and positioned.

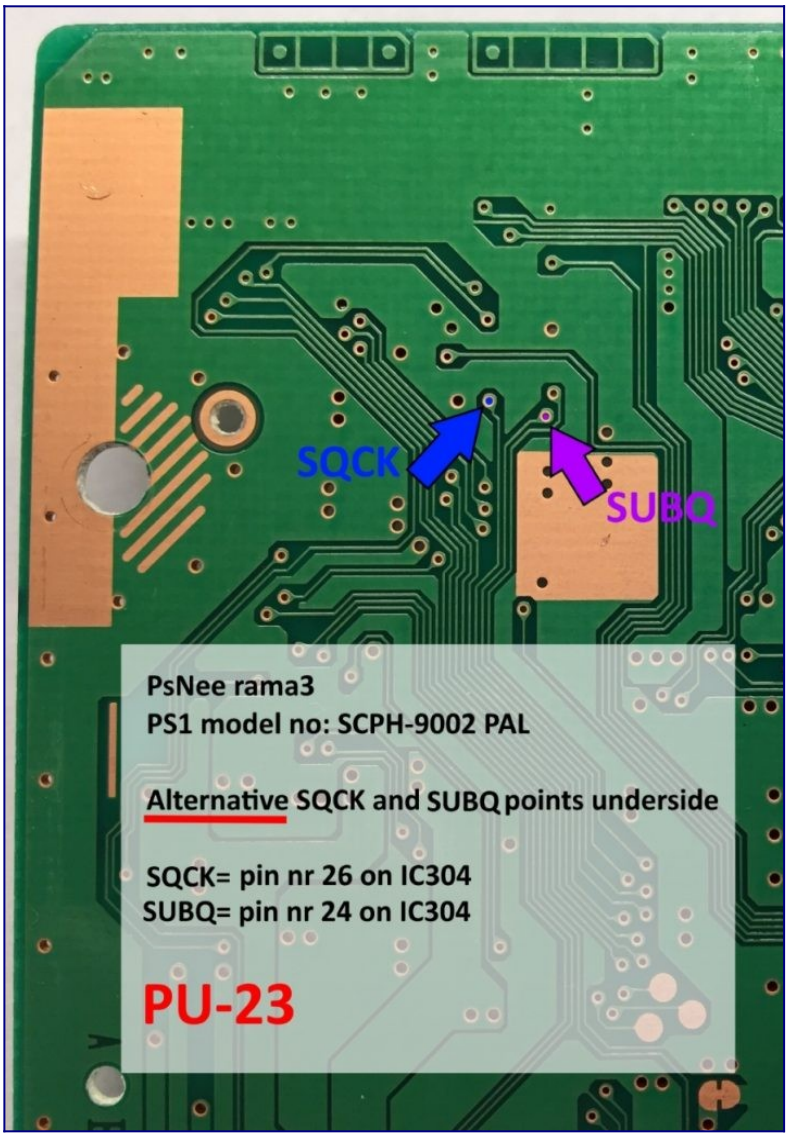


PU-23 PsNee modchip installation diagram

The last version of the original PlayStation 1 design was the SCPH-900x. Most of these consoles have PU-23 boards inside. The most notable difference between the 9000 series and earlier consoles is that Sony removed the parallel IO port on the back. They also shrunk down the size of the board significantly.

PU-23 PsNee modchip installation diagram





Above are some diagrams for installing the PsNee into a PU-23. The second shows some alternative points.

About the points

For models SCPH-xxx1, SCPH-xxx2, you can ignore the RESET. On the other hand, the model and SCPH_xxx0, you will have to use the RESET point, and the BIOS points.All of the points are fairly easy to solder to

Installation tips

Here are some tips I have for you when you are soldering your chip into the PU-23.

- **Cut your wires to be as short and direct as possible.**
- You don’t need to connect pins one and two of the ATtinyX5 chip. Just desolder the wire.

- Use a multimeter to probe around for alternative VCC and GND points closer to where you position your chip for a cleaner installation.
- For DATA, be careful not to bridge the connection to the other resistor next to the one you are soldering to.
- Don’t apply too much solder to SQCK, or you’ll bridge the pins on the chip.
- For the wires that go into the holes, or vias, of the board: it’s easiest to stick a small 30 AWG wire through the hole, then heat the wire and hole while adding solder.
- Placing the chip on top of the chip above SUBQ and SQCK is a good spot.

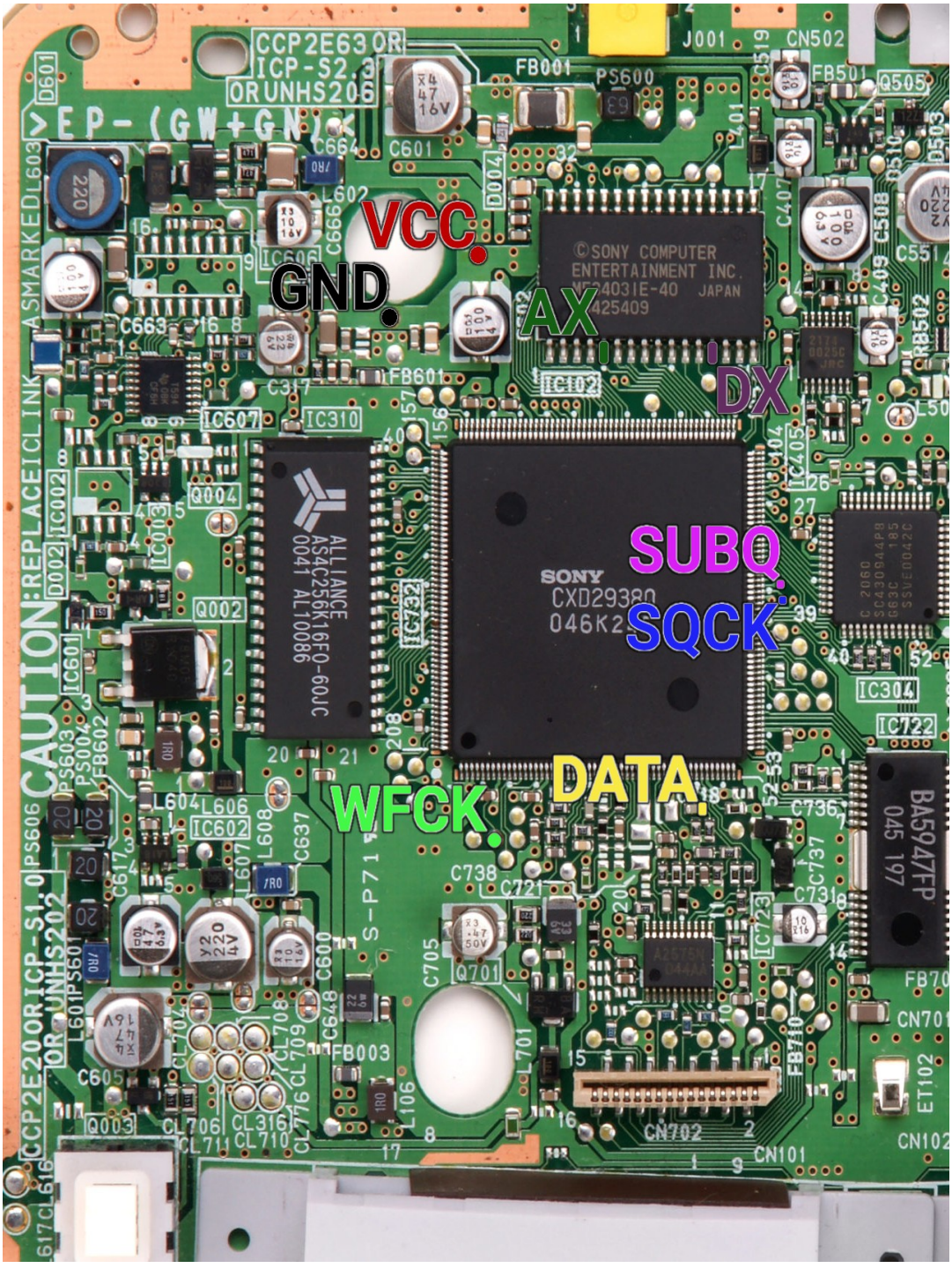
Example installations

This section has photos of some successful installations which you can use to get a better understanding of how everything is wired and positioned.

PM-41 PsNee modchip installation diagram

.Towards the end of the PlayStation 1's life Sony released a slim model called the PSone, or SCPH-10x. Many of these systems have PM-41 boards inside, which is what this page covers. Some of the boards are PM-41 (2) and require you to use a different diagram. The model is printed on your board if you take apart your console.

PM-41 PsNee modchip installation diagram



About the points

For models SCPH-101,SCPH-103, you can ignore the RESET. On the other hand, the model SCPH-102and SCPH-100, you will have to use the RESET point, and the BIOS points.All of the points are fairly easy to solder to

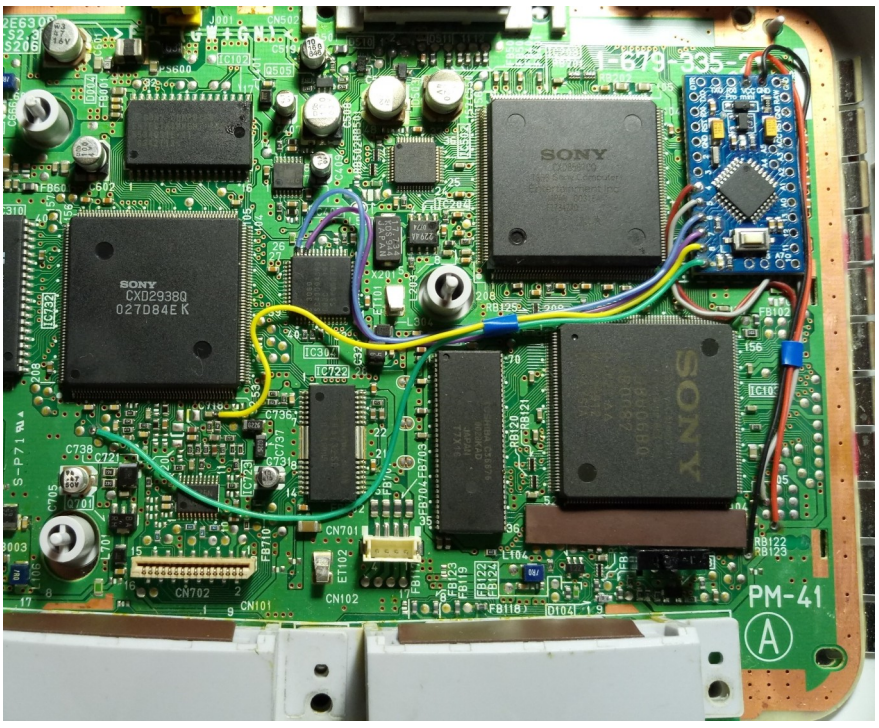
Installation tips

Here are some tips I have for you when you are soldering your chip into the PM-41.

- **Cut your wires to be as short and direct as possible.**
- You don’t need to connect pins one and two of the ATtinyX5 chip. Just desolder the wire.
- If you are using an ATtinyX5, or have a SCPH-101, then ignore pins A18 and D2.
- Use a multimeter to probe around for alternative VCC and GND points closer to where you position your chip for a cleaner installation.
- For the wires that go into the holes, or vias, of the board: it’s easiest to stick a small 30 AWG wire through the hole, then heat the wire and hole while adding solder.
- For DATA, be careful not to bridge solder to a nearby component, or knock the capacitor off the board.

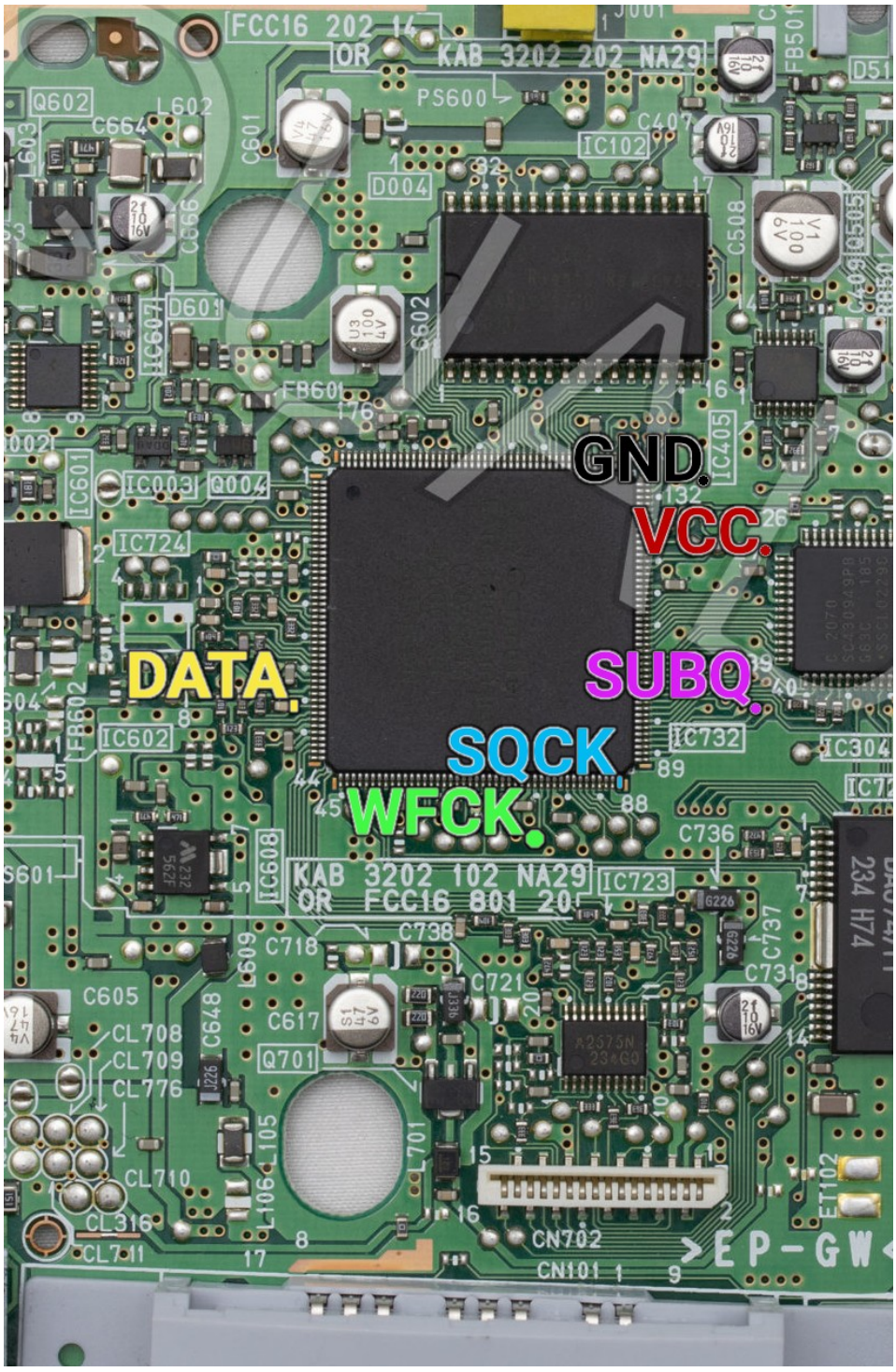
Example installations

This section has photos of some successful installations which you can use to get a better understanding of how everything is wired and positioned.



PM-41 (2) PsNee modchip installation diagram

Towards the end of the SCPH-10x PSone production Sony revised the board slightly and released the PM-41 (2) board. This page covers the installation diagram for installing a PsNee chip into the PM-41 (2).



PM-41 (2) PsNee modchip installation diagram

Above is the installation diagram. Just match each colored pin label in the diagram with any matching colored point on the board.

Installation tips

Here are some tips I have for you when you are soldering your chip into the PM-41 (2).

- **Cut your wires to be as short and direct as possible.**
- You don’t need to connect pins one and two of the ATtinyX5 chip. Just desolder the wire.
- Use a multimeter to probe around for alternative VCC and GND points closer to where you position your chip for a cleaner installation.
- There may also be easier to solder to points for SQCT and SUBQ.
- Be super careful when soldering SQCT and SUBQ. It’s easy to bridge pins if you aren’t careful.
- Be careful with the DATA connection, it’s easy to accidentally knock the capacitor off the board.

Guide disclaimer

I’m not a PlayStation 1 modchip expert. This guide has been put together from a combination of my experience, and what I have found online. Many of the sources for PlayStation 1 modding information have gone offline, or are spread out across many forum threads.

My goal is to have a single guide that covers as much information as possible, making it easier for everyone to get solid information about PlayStation 1 modchips. If you have any suggestions, corrections, or additional information, please leave a comment and I’ll be happy to update this guide.

Modchip specific guides

Here’s a list of modchip specific guides, which include links to installation diagrams for each board, and source code to make your own chips.

- MM3 modchip
- Mayumi v4 modchip
- ONEchip modchip
- PsNee modchip

Modchips that aren’t in the list above are covered in less detail in a guide covering other older modchips.

What is a modchip, and what is it used for?

The idea of a modchip is pretty straightforward. It is typically a small chip or board that is soldered to various parts of a video game console. The chip is programmed with special software that is used to circumvent DRM protection on the console.

In the case of the PlayStation 1 a modchip allows the console to do many new things. Here’s a list of MM3 and Mayumi v4 modchip features.

- Play video game backups from any region.
- Play genuine games from any region.
- Works with all games, even the ones with anti modchip code.

A common use of a modchip is to play games from other regions, like Japan, that weren’t released in the United States. With a modchip there is no need to import a console from each region.

PlayStation 1 modchip history

The original PlayStation was one of the first consoles to have a modchip available for it. This was primarily due to the rise of cheap CD burners and blank discs that allowed game backups to be easily produced. That and the rising popularity of downloadable game backups online.

PlayStation 1 modchips were produced as early as 1996, not very long after the console was first released. These early mod chips were very expensive, but over time they were reverse engineered, cloned, and sold prices fell as a result. The original chips used the PIC16C54 microcontroller.

In 1997 a new modchip was released called “Old Crow”. This new modchip was another clone of the original modchip, but the source code was released to the public, reducing the cost of the chip even further. “Old Crow” was originally made for the Zilog Z8 microcontroller, but versions were released for the PIC16C54, and PIC16C84 as well.

Later in 1997 the “Old Crow” code was ported to the then new PIC12C508 microcontroller, which is still being used for modchips today. The new chip dropped the modchip prices to the price range they are still sold at today.

Currently there are a few options for the PS1 modchip. The three most popular chips today are the MM3, the Mayumi v4, and the ONEchip.

All three are based off of the PIC12C508, but there are ports that allow the code to run on other chips like the PIC12F629. These chips are based on the “Old Crow” code with additional features like support for anti modchip games.

MM3 and Mayumi v4 chips are very similar, but operate slightly differently. Because of this some consoles may work better with one than the other.

ONEchip modchips are designed specifically for PAL region PSone’s, and don’t work with anything else. They were designed to bypass additional protection inside the SCPH-102.

There is also newer code designed to be used with modern Atmel processors like the ATmega and ATtiny series of chips.

How does a PlayStation 1 modchip work?

PlayStation 1 modchips were designed specifically to bypass the copy protection and region coding that was built into game discs to prevent piracy.

In particular there is a stream of data sent from CD drive to the processor every time a game starts. The steam is sent in serial at 250bps and includes the text SCEI for Asian discs (NTSC/J), SCEE for European discs (PAL), or SCEA for North American (NTSC/US) discs.

The console’s processor is programmed to look for this stream and only accept discs from its region. To bypass this copy protection mechanism the modchip removes the signal from the disc, and replaces it with the correct region coding. So for an American console the modchip will send the processor the SCEA string regardless of the disc that is inserted into the system is coded to send.

Anti modchip games have additional data in this stream to throw off some modchips. Newer chips like the MM3 check for this additional data and let it through to the processor if it exists, along with the spoofed region coding string.

PAL PSones have additional protection built into the BIOS. ONEchip and PsNee modchips connect to a couple BIOS pins and patch it to allow booting out of region games. Without the BIOS patch

PAL PSones can only play PAL region games. Japanese systems starting with some SCPH-3000 models also have a similar region lock, but it hasn't been bypassed by any modchips yet.

If you want to learn more about how a PS1 modchip works, I recommend taking a look at the PsNee project on Github. The project is the modchip code for Atmel processors. The code is well documented and explains how everything works.

What are the limitations of a modchip?

- PAL and NTSC incompatibility with your TV

In North America we have NTSC-U/C consoles and TV's, and in Japan there are NTSC-J consoles and TV's. Meanwhile Europe and some other parts of the world there are PAL consoles and TV's. In the case of the PlayStation 1 the modchip can be used to unlock the console's ability to output video for any region based on the game being played.

The problem with this is that not all TV's accept all input signals. For example if you put a PAL game into your modded system and try to play it on an American TV you might not get any video on your TV.

To get around this problem I recommend buying a cheap composite to HDMI converter. Typically these will take the weird analog signal that may be associated with a game from another region, and convert it into a nice and standard HDMI signal that your TV will display without any problems.

- Japanese consoles and video glitch with PAL games

Japanese consoles with specific clocks will produce bad video signal there are mods to replace the crystal with a pulse generator https://github.com/L10N37/PSXTAL for more information.

Table des matières