Pokémon Stadium Color Changes

By Fëanen (Twitter: @feanen_atreides)

Latest update (9/4/20): Added preliminary data about Japanese character encoding.

Click here to go directly to the gallery

One of the most unique but poorly understood parts of the Pokémon Stadium games is that nicknamed Pokémon can be different colors than usual. Sometimes the change is subtle, but it can also be very striking – a Pikachu with yellow-green fur, a dark blue Bulbasaur with yellow eyes, or a blue-tongued Rhyhorn! I've seen a few partial attempts to explain how the games determine the color - a lot of people think that the nickname should be some sort of variation of the original name (APE-PRIME, poliWRATH) or that any two of the same Pokémon with the same nickname will be identical. These ideas seem to be based on how NPC Pokémon are nicknamed, but a bit of experimentation showed me that it isn't the case at all. So, what is it that makes a Pokémon become a different color?

Full disclosure – I'm not a programmer or hacker, and what I know about computer science mostly comes from YouTube videos. Fortunately, those videos taught me about the two major programming factors that go into the color changes – hexadecimal numbers and character encoding. Most of my research was 1% inspiration, 99% perspiration – trying a bunch of nicknames and different OTs on Pokémon until I discovered a pattern.

The Name Factor

What is it that determines what color a Pokémon will become? The short answer is it's found by combining a Pokémon's nickname, the original trainer's (OT) name, and the OT's Trainer ID number. Since the OT and not the *current* trainer is the important factor, Pokémon will never change color when traded.

All digital data is represented by a numerical value, and in Pokémon Stadium (and other 8-bit character encodings) every possible character is represented by one byte, ranging from 00 to FF (255) in

hexadecimal. Each letter in the Pokémon's nickname and OT name is represented by one of these bytes. Unlike the Game Boy games, Pokémon Stadium encodes characters with a common system called Extended ASCII. Not every character is in the game, and there are four special symbols ($^{P}_{K,}$ $^{M}_{N}$, ♂ and ♀) that replace unneeded characters. The characters available in Stadium and their hex values can be found here:

https://github.com/Guernouille/Stadium_Editor/blob/master/sources/initialize_char_table.cpp#L23

Not all of these characters can be used in names – at least not without cheating. In fact, Team Rocket Grunts have Pokémon with numbers in their nicknames, which players can't do in the first two generations. I guess it's not surprising that the bad guys don't play by the rules! Although Stadium will work with "illegal" nicknames, for unhacked Pokémon the letters and symbols available in the GB games are what matter. I've made a table of these characters in their English Gen 1 order, plus six more only available in German games, along with their hex values. The blank cell (20) represents a space.

A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**	**L**	**M**
41	42	43	44	45	46	47	48	49	4A	4B	4C	4D
N	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
4E	4F	50	51	52	53	54	55	56	57	58	59	5A
a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
61	62	63	64	65	66	67	68	69	6A	6B	6C	6D
n	**o**	**p**	**q**	**r**	**s**	**t**	**u**	**v**	**w**	**x**	**y**	**z**
6E	6F	70	71	72	73	74	75	76	77	78	79	7A
	×	**(**	**)**	**:**	**;**	**[**	**]**	$^{P}_{K}$	$^{M}_{N}$	**-**	**?**	**!**
20	D7	28	29	3A	3B	5B	5C	BC	BD	2D	3F	21
♂	♀	**/**	**.**	**,**	**Ä**	**Ö**	**Ü**	**ä**	**ö**	**ü**		
A9	BE	2F	2E	2C	C4	D6	DC	E4	F6	FC		

♂ and ♀ can only be used in Gen I. Blank spaces and ; can't be used in Gen 2 German games.

The trainer ID is a 2-byte number in hexadecimal, and when you add the hex numbers for all the letters in the OT name, the nickname, and the two bytes that make up the ID number you get what I call the Name Value (NV). Only the last 2 digits of the NV count – the rest can just be ignored. All NVs will be given in hexadecimal.

Here's an example. Let's say your Pokémon's nickname is Ted, the OT is Dan, and the Trainer ID is 12345. Here are the hex values – remember that the 2 bytes for the ID are treated as separate numbers and added together:

Dan: 44 61 6E

Ted: 54 65 64

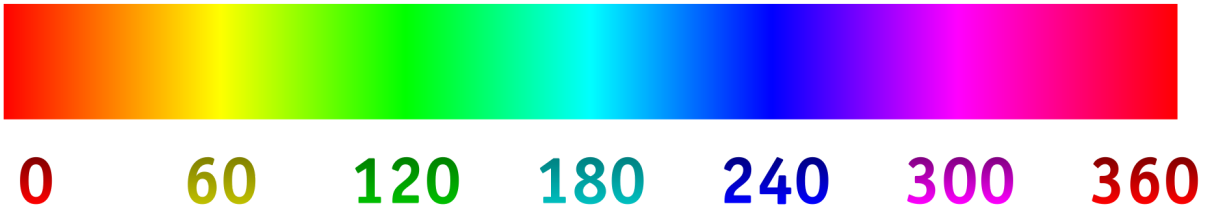
12345: 30 39

Add up all 8 of those numbers in a hex calculator, and you'll get 299. Since only the last two digits matter, Ted's NV is 99.

Color Modifications

Now that we know how nicknames and trainer data are combined to create a special value for each Pokémon, let's look at the mechanics of the actual color changes. Boxes full of the same Pokemon with different nicknames showed me that there were too many possible colors for each Pokémon to have specific palettes assigned – there had to be a formula that automatically adjusts the color. One key discovery was that the RGB values of a Pokémon's texture aren't directly changed by the formula – instead, the system changes their hue.

A lot of people are used to RGB colors – mix the three primary colors to get the result – but the HSL (hue, saturation, lightness) color space works a bit differently. Colors are arranged by 360 degrees like so:



Hue scale (From Wikipedia)

What matters for us is this – if you alter the hue of an entire image or texture, all the colors will change in a predictable way. Increases move every color forward on the wheel and decreases make all the colors go backward. A change of +180 will move every color halfway across the color wheel – reds become cyan, blues become yellow, etc. So, if your Pikachu's color changes by +15, its yellow fur is going to become greener, but if its color is -15 the yellow will become more orange. The saturation and lightness components are never changed for nicknamed Pokémon. One huge benefit of modifying hues instead of RGB values is that white and black parts of textures don't change.

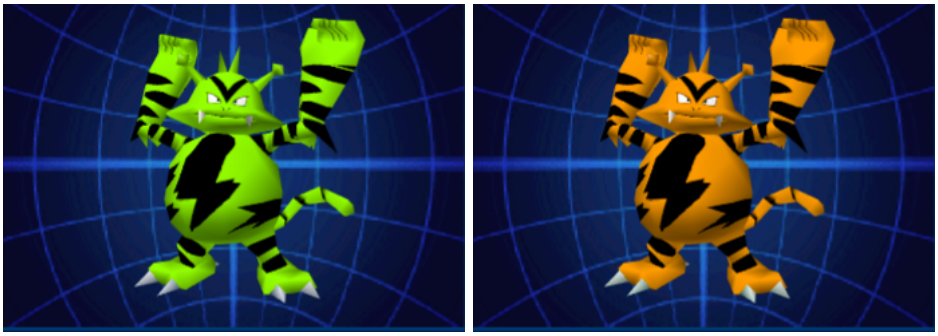
All Pokémon have a minimum and maximum hue. Gen 1 Pokémon usually have a bigger range of hues than Gen 2 Pokémon, probably because by Stadium 2 shiny Pokémon were added to the mix. The range varies by Pokémon, but it isn't usually more than +/- 60.

Here's where the NVs come in. Each Pokémon has an approximate NV in the middle of its range where it's at its normal colors – for Bulbasaur this is around 64, while for Celebi it's around A2. Higher NVs than this increase the hue, while lower NVs decrease it. NVs of 0 or FF produce the minimum and maximum hues, respectively.

A very recent breakthrough came when hacker jrra released a patch that re-implements the debug menu of Pokémon Stadium 2. By the time I got this patch I'd already discovered that there were hue ranges and had some rough ideas on what they were for certain Pokémon. The debug features fill in a critical missing piece: among other data and tools, the model viewer has the exact hue ranges for every Pokémon! The ranges for each Pokémon are in columns 3 and 4 of this document. Here's some examples:

Venusaur: -30, + 40

Electabuzz: -20, +30



Vaporeon: -20, +50

Elekid: -5, +10

Notice how even though Elekid and Electabuzz are related and have the same colors, Electabuzz has a much bigger range? That's because they come from different generations. Usually, only Pokémon who change their color scheme when they evolve (like Eevee or Bulbasaur) have significantly different hue ranges in the same generation. Others, like Clefairy or Zubat, are either the same or close. Minimum

and maximum hues stayed the same between games, so if you use the same Pokémon in Stadium 1 and then hook the game cart up to Stadium 2 it shouldn't look any different. The only exception I've found is Poliwag, which has the same hue range in Pokémon Stadium 2 but a new mouth texture that changes a lot more intensely – it can be blue or green. Poliwag's menu icon wasn't changed in Stadium 2, so it still shows what it would look like in the original.

There are a few Pokémon with extremely high minimum and maximum hues – in fact, they go all the way through the color wheel. These are Pokémon who are either grey or a very dark color – Rhyhorn, Rhydon, Geodude, Graveller, Onix, Magnemite, Magneton, Grimer and Muk. Yes, Grimer and Muk – they're a darker purple in the Stadium games than in other media. I call these "full-spectrum Pokémon." For these Pokémon, NVs of 0 and FF are identical: the game assigns these values to -180 and +180, which are the same thing in a 360-degree wheel.

The funny thing is that while these big color ranges don't always affect a Pokémon's body too drastically, their tongues and sometimes eyes are more vivid and go through the entire spectrum – you can have a Rhyhorn with a lime green or cyan tongue. Onix was a big surprise – if you look closely at its eyes, you'll see that it has colored irises that go through the spectrum. It's a bit small to make out on the rendered model, but the actual texture is quite vivid.

Decoding the Formula

The only way to know for sure how the games translate NVs into hue changes would be to look at the code. Until somebody does that, I've found a formula that seems very accurate. If math's not your thing, feel free to move on to the next section. Let's start by defining our basic variables. For these formulas, we treat the minimum hue as being positive instead of negative.

M = Minimum hue
X = Maximum hue
N = Name Value

The next thing we need is the Pokémon's range of hues (R). This is easy to find.

$$R = M+X$$

Next, we determine what portion of a Pokémon's range a single NV represents. I call this the Increment (I).

$$I = R/255$$

Now we need to find out what NV (in decimal notation – we can convert to hex later) corresponds to the Pokémon's default palette (D). The result isn't necessarily a usable NV, since it can have a decimal point and NVs are always whole numbers

$$D = (255/R)*M$$

At last, we come to my formula for finding the Pokémon's hue (H)

$$H = (N–D)*I$$

When N>D, we get a positive value and the Pokémon's hue moves toward the maximum. When N<D, the value is negative, moving the hue toward the minimum. Comparing textures taken with the model viewer (which allows you to view any whole-number hue adjustment) with textures for various NVs shows that this formula produces results that are pretty darn accurate -most NVs aside from 00 and FF will give a result with a decimal point, so some NVs produce hue changes between the ones that the model viewer can show. These slight distinctions aren't usually visible to the naked eye.

For anybody who wants to experiment, I've created a downloadable spreadsheet with my formulas. Be sure to save it to your own Google Drive or device, as the original can't be edited by other users. Most of the important variables (Min, Max, Range, Increment, Default NV) are already calculated. The first page is the hue calculator – find the row with the Pokémon you want to look up, enter its hexadecimal NV in column H and it will give you its approximate hue. On the second page, you can enter the hue you want in row G and columns H and I will give a corresponding NV in decimal and

hexadecimal. The results won't be perfect since the NV has to be a whole number, but the difference between the desired and actual hue change will be too small to matter.

Japanese-Language Games

When Pokémon Stadium came out in the West, what we really got was Japan's Pokémon Stadium 2 (our Stadium 2 was called Stadium GS in Japan). Their original Pokémon Stadium (which I'll be calling Stadium J) was a pretty stripped-down game with a limited roster of playable Pokémon, so overseas fans didn't miss much. However, every Pokémon has a model that you can view in the Pokédex and boxes, and the nickname-color system is already implemented.

The Stadium series uses a character encoding system called EUC-JP. Since Hiragana and Katakana are encoded into two bytes, these have to be added together. For example, the character ア is represented in hex as A5A2, so you add A5+A2 to get 147 (or just 47, since only the last 2 digits of a number ever matter for NVs). All hex values for this system of encoding can be found here, and a listing of characters which are legal for OT and nicknames can be found here. Gen I and II Pokémon games had a slightly quirky way of handling Japanese characters, so there are a few things that need testing: most importantly, a few hiragana and katakana that are similar-looking are treated as being the same character by the GB. I do not know whether the Stadium games treat these characters as hiragana or katakana at this time.

Even without hex values for the characters, checking the colors is easy. Using PKHeX, I took a Japanese save file and set every Pokémon's nicknames and OTs to be blank, which gives them a value of 0. This way, the NV for each Pokémon is determined by the TID, which can be set from 0 to 255 to see the range of available hues. This trick works on the Western games too, but I've never tested what happens in the actual GB game when you have blank name fields, so proceed with caution!

The one Pokémon who's the most different in Stadium J is Dewgong. In the other games it has a pretty normal range, but in Stadium J it's actually full-spectrum! The fur color does change, but like other full-spectrum Pokémon the tongue and eyes are a lot more prominent.

Shiny Pokémon

Shiny Pokémon were added to Stadium 2: they stay the same color whether they're nicknamed or not. Since hue changing was already part of the game's mechanics, the programmers used it for shiny Pokémon too. Shiny Pokémon's colors are mapped to a specific HSL alteration which can be seen in the debug model viewer. Saturation and lightness are actually used this time, so shiny Pokémon can be more or less vivid than normal ones.

Four Pokémon (Weedle, Kakuna, Geodude and Graveler) have changes that are also possible with a nickname, making their N64 shiny colors a bit less special. Eight Pokémon (Gyarados, Noctowl, the Clefairy line and the Jigglypuff line) have special textures for their shiny forms instead. The HSL numbers for shiny Pokémon can be found in the same document as the hue ranges for nicknamed Pokémon. A while back, Deviantart user icycatelf took Gen 3 sprites and applied their Stadium 2 shiny colors – check it out here. A lot of those were based on screencaps of all 251 shiny Pokémon, but the link to them no longer works.

Pokemon Eggs

In Stadium 2, Pokemon eggs change hue as though they're the species they're going to hatch into, with the OT data of whoever bred them and the nickname EGG (hex:D3). Shiny eggs always use the default palette, which can be useful for shiny hunting. A selection of differently hued eggs have been placed in the gallery, with file names indicating the +/- change. Huge thanks to Tama for bringing egg colors to my attention! Check out her excellent video on using egg colors to shiny hunt here.

Smeargle

On top of having a hue range, Smeargle's tail and back marking can be red, green or blue, for nine possible color combinations. This is determined by a factor called the 'Sub ID" in the game, and it's based on Smeargle's DVs. Here's how it works:

Take the single-digit hex values of Smeargle's Special Attack, Defense, Speed and Attack and put them in a row in that order. This will give a 2-byte number. Let's say the values were 2, 14, 7 and 12; this would be written as 2E7C. Divide this by 9 and the remainder is the Sub ID. A quicker way to do it is converting the hex value to base 9 here and using the last digit as the result – for our example, the hex number is equivalent to 17282 – its Sub ID is 2. The Sub IDs correspond to the colors as follows:

Sub ID	Tail	Back
0	Red	Red
1	Green	Red
2	Blue	Red
3	Red	Green
4	Green	Green
5	Blue	Green
6	Red	Blue
7	Green	Blue
8	Blue	Blue

Our example Smeargle will have a blue tail and a red paw print on its back. If it's nicknamed, this will be modified by whatever hue its NV gives. Shiny Smeargle's hue change is +180, making the red, green and blue become turquoise green, indigo and orange. Since shiny Pokémon are determined by specific IVs, shiny Smeargle can only have a Sub ID of 2, 3, 4, 6, 7 or 8.

The Gallery

Taking pictures of all 255 NVs for 250 Pokémon is obviously out of the question, and most of them would be almost identical anyway. Instead, I've taken screencaps of each Pokémon's minimum, standard and maximum hues. I've put them all in a Google Drive folder, divided by generation.

Pokémon are organized by their National Pokedex numbers with at least three pictures representing the minimum, normal and maximum colors. A few Pokémon with very large ranges got an extra picture or two to capture their spectrum better. Each filename ends with its NV in decimal notation, or "N" if it's the default color. For full-spectrum Pokémon, I've taken every 16th hue starting with an NV of 0F; 127 represents the standard coloration. One image of Unown is included for completeness. One of my absolute favorites is Nidoqueen 255 – if you ask me, it would make a much better shiny for her than that green palette we got.

Just for fun, I took some art, sprites and models from other games and applied hue changes (mostly min, max and shiny) to see what they'd look like. I even included a couple Mega Evolutions as a taste of what might have been. These can be found in a folder called "Modified Art." Maybe this will inspire other fans and artists to have some fun with Pokémon Stadium's unusual colors!

My screencaps were mostly taken using the PC in Professor Oak's Lab, which is good for viewing each Pokemon in your boxes very quickly. Sending the Pokémon out in battle gives a better view, but it also takes a lot longer – with over 750 pictures to take, it would be quite a project! My screencaps were mostly taken before I had access to Stadium 2's model viewer, which gives the best view of each Pokémon, but has a big flaw for screencap taking - the icon for the Pokémon is always overlaid on the screen, and unless you zoom out it covers part of the model. I did end up using the model viewer for Jumpluff, since I couldn't get its fluff to render right on the PC. It doesn't match the rest of the gallery, but at least it's correct.

I've tried to find a pose that captures each Pokémon's colors well – since I was dealing with moving models, the pose for different NVs isn't always quite the same. The pictures were a lot easier to take in Stadium 1, where you can pause and rotate Pokémon models when you view them on the PC. Unfortunately, Stadium 2 just constantly turns the model in a circle, and there's nothing you can do to

manipulate it. Most of the coolest changes are to Gen 1 Pokémon, so having a more limited view in Stadium 2 isn't a huge loss.

A handful of Pokémon needed to be shot from multiple angles to capture all their colors: I've put these screencaps in folders called "Extra Poses" for each generation. Golbat and Skarmory never spread their wings on the PC, so I used other modes to get these poses. I've included Stadium J Dewgong and Poliwag's min and max hues from Stadium 2 in the Gen 1 folder.

I used Stadium revision 1.1 for these screencaps, so Jynx has a purple face. Since black textures are never affected by altering the hue, I didn't see any point in documenting the changes to her original (and understandably controversial) palette.

Notes and Oddities

- When you apply the HSL modifications to a default texture in GIMP, the results are always slightly off from what the N64 gives. I think the N64's system for converting RGB to HSL and back may be less precise than GIMP's. The differences aren't visually significant.
- Gastly's icon shows the gas part of its body changing color, but on the model only the face changes – the cloud around it is always the same purple. Weirdly, the gas *does* change on shiny Gastly, which has a hue modification of -124.
- Unown is *technically* the only full-spectrum Gen 2 Pokémon, but since its palette is 100% greyscale the color will never actually change. It doesn't have a shiny color or texture defined either.
- Almost all Pokémon's hue ranges are multiples of 5. The only exceptions are Noctowl (+7, -7) and Sentret (+14, -15). I have no idea why these two are different from the rest.
- ??? (an empty Pokémon slot in Gen 2, represented as a Substitute doll) is coded to be full-spectrum. Actual Substitute dolls don't change color when used by a nicknamed Pokémon.

Resources Used

- PKHeX – The ultimate Pokémon save file editor. I honestly couldn't have done my research without it!
- jrra's Zone - Pokémon Stadium 2 – The patch that restores the debug menu, plus explanations of its features.
- GIMP – A popular free image editor I used to inspect the HSL values of individual pixels in textures and icons.
- isleep2late's Crystal to Crystallize all Crystals.SAV – A save file with all 251 shiny Pokémon. After modifying the data so the Pokémon weren't shiny, I used this save as the basis for most of my Stadium 2 experiments.
- Project 807/153/400 – Save files with all Pokémon for every game in the series. The RGBY (Japan) save was used to test Stadium J.
- A 151 species save for Pokémon Red with an OT of ROG that I used to test colors in Stadium 1. Unfortunately, for some reason I lost the bookmark and can't find where I got it from. If anybody's familiar with this, please let me know so I can give proper credit!
- Bulbagarden Archives – Source for original images used for my modified art and sprites.

.