

nesdoug

Sprites, Again

Ok, with neslib, you will be loading the OAM (sprite) buffer. It will automatically send them to the PPU during v-blank. The only thing you need to do, is keep track of the index into the OAM buffer.

```
unsigned char sprid;
```

Oh, and I clear the OAM at the top of every frame.

```
oam_clear();
```

(Alternatively, you could call `oam_hide_rest()`; at the end of all sprite drawing code.)

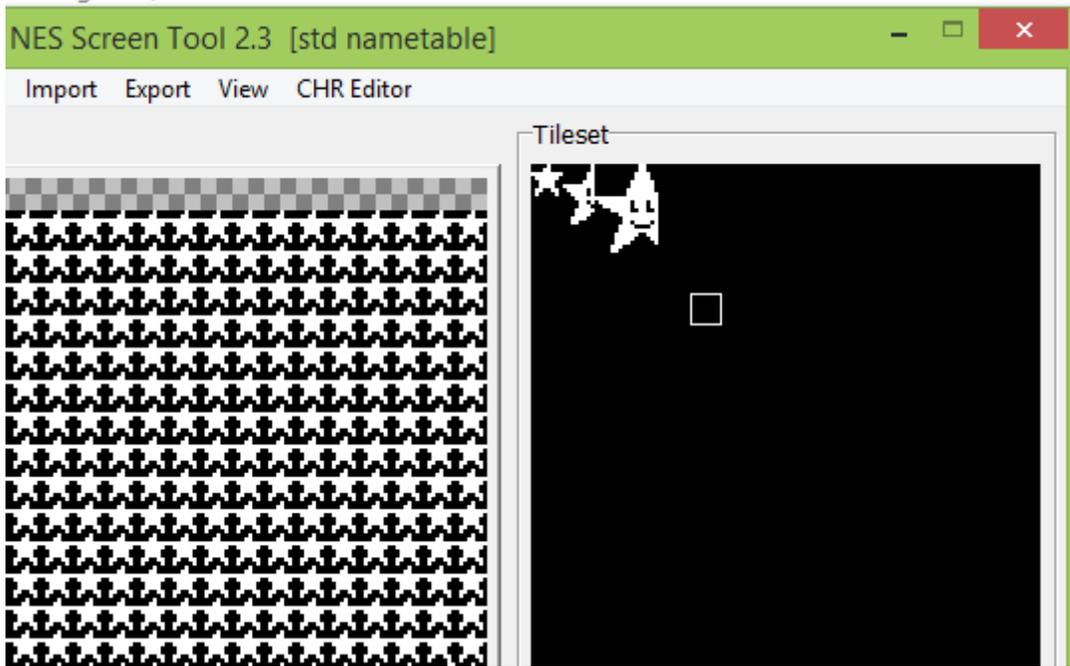
To send 1 sprite, you use this function.

```
oam_spr(unsigned char x,unsigned char y,unsigned char chrnum,unsigned char attr,unsigned char sprid); // it returns a char, the current index #
```

```
sprid = oam_spr(X_position, Y_position, 0, 0, sprid);
```

`chrnum` is the tile #. `attr` is which palette 0-3 (and H and/or V flipping).

I made some metasprites with NES Screen Tool 2.3. And used the 'Metasprite/Put single metasprite to clipboard as C' option, and pasted it into a .c file. Notice that I only made the left half of the sprite graphics, because the right half is just a mirror image of the left, and you can flip sprite tiles.



Then, to put a metasprite into the OAM buffer...

```
oam_meta_spr(unsigned char x,unsigned char y,unsigned char sprid,const unsigned char *data); // it
returns a char, the current index #
```

```
sprid = oam_meta_spr(X_position2, Y_position, sprid, metasprite);
```

...where 'metasprite' is an array of chars that represents the sprites used, and their relative positions.

So, here's the code. I'm shifting the position of each sprite object by adding 1 to it's Y position each frame. You use `ppu_wait_nmi()` to wait for the beginning of the next frame.



<http://dl.dropboxusercontent.com/s/92m5gikr51emf9d/lesson24.zip>
 (<http://dl.dropboxusercontent.com/s/92m5gikr51emf9d/lesson24.zip>)

[August 9, 2017](#)[August 10, 2017](#) [dougfraker](#)

[Create a free website or blog at WordPress.com.](#)