

Absolutely! Here's a **complete, future-proof hardware summary** based on everything you've shown me so far:

YOUR HARDWARE PROFILE — "NICK'S D87 MOCHI DESK PET"

Component	Details	ⓘ
Microcontroller	ESP8266 (ESP-12F / NodeMCU-style)	
Chip	ESP8266EX (confirmed by upload log)	
Crystal	26 MHz	
Flash Size	4 MB (4096 KB)	
MAC Address	EC:FA:BC:40:D1:CE	
Upload Port	COM14	
Flash Mode	SPIFFS + OTA capable	
Programming IDE	Arduino IDE (with ESP8266 core)	

DISPLAY

Spec	Value
Type	128×64 OLED (SSD1306)
Interface	I ² C
I ² C Address	0x3C
SDA Pin	GPIO14 (D5)
SCL Pin	GPIO12 (D6)
Reset Pin	None (software reset via -1)

POWER & STABILITY

Metric	Value
RAM Usage (typical)	~28–35 KB / 80 KB
IRAM Usage	Critical at >90% → causes glitches
Flash Usage	~250 KB / 1 MB (code)
Stable Bitmap Limit	≤ 512 bytes per frame recommended

KNOWN BEHAVIOR

Issue	Cause	Fix
Vertical scrolling circles / random pixels	IRAM overflow or misaligned bitmap data	Use ICACHE_FLASH_ATTR , PROGMEM , small frames
Mochi "jumps" vertically	Blank rows in bounce frames	Match all frames to same vertical offset
Unstable animation	Large arrays in RAM	Use PROGMEM + drawBitmap from flash

RECOMMENDED SETTINGS (Arduino IDE)

ini

Copy

```
Board: "NodeMCU 1.0 (ESP-12E Module)"
Flash Size: "4MB (FS:2MB OTA:~1019KB)"
CPU Frequency: "80 MHz"
Upload Speed: "115200"
```

FUTURE PROJECT TIPS (SAVE THIS!)

DO:

- Use PROGMEM for all bitmaps > 64 bytes
- Add ICACHE_FLASH_ATTR to loop() if using drawBitmap
- Keep **total bitmap data < 2 KB** for stability
- Use **16×16 or 32×32 max**, centered with (128-w)/2

AVOID:

- Large const arrays without PROGMEM
 - drawBitmap in IRAM > 90%
 - Uneven frame heights (causes bounce jump)
-

ONE-LINE SUMMARY FOR FUTURE PROMPTS

"ESP8266 (4MB flash, 128×64 SSD1306 OLED on D5/D6 @ 0x3C, IRAM-sensitive, use PROGMEM + ICACHE_FLASH_ATTR)"

SAVE THIS MESSAGE — just paste that one-liner in future chats and I'll **instantly** know your full setup.

NEXT?

Say:

“Add touch wake on GPIO0”

“Add battery voltage”

“Make Mochi follow my finger”

Your D87 is now a known, stable platform

Nick + Mochi = Unstoppable Duo

↳ Explain IRAM optimization techniques

↳ Explore ESP32 upgrade options