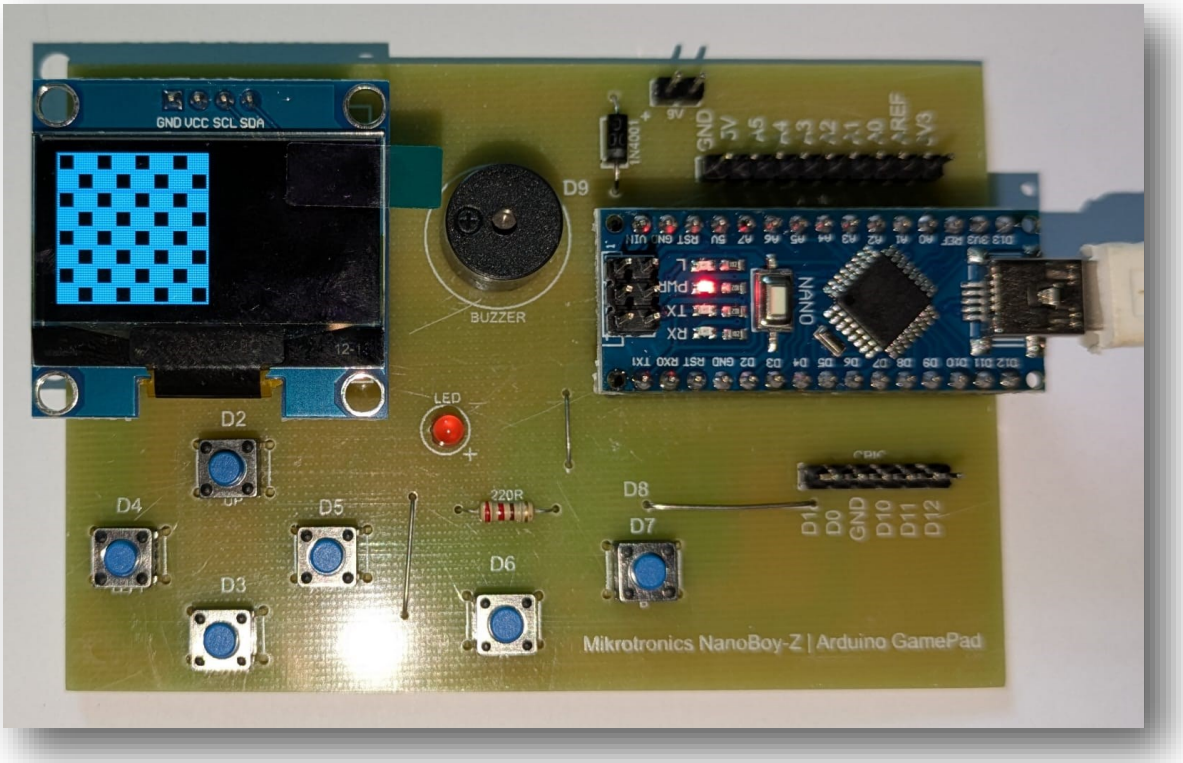
NanoBoy User Manual

(First Edition – 2025, Mikrotronics Pakistan)



1. Introduction

NanoBoy is a small handheld console designed by Mikrotronics Pakistan to help students and hobbyists learn embedded programming in a fun, game-oriented way.

It is built around an **Arduino Nano** and includes:

128×64 pixel OLED display (SSD1306 or SH1106)

Six input buttons (Up, Down, Left, Right, A, B)

User LED

Piezo buzzer

NanoBoy comes with its own **NanoBoy Library** that makes graphics, sound, and input simple. With just a few lines of code you can draw shapes, print text, detect button presses and even build games.

Download the library and examples from our GitHub page:
https://github.com/ameriqbalqureshi/NanoBoy

2. Hardware Details

Microcontroller: Arduino Nano (ATmega328P)
Display: OLED 0.96" (SSD1306) or 1.3" (SH1106), I²C interface

Buttons:

- D2 = UP
- D3 = DOWN
- D4 = LEFT
- D5 = RIGHT
- D6 = A
- D7 = B

LED: D8
Buzzer: D9
OLED I²C: SDA = A4, SCL = A5

Power Supply

NanoBoy is powered by USB (5 V). You can also power it via the Nano's VIN pin (7–12 V). There is a header with reverse polarity protection where you can give 7-12V and it will get regulated power supply through Nano's Vin pin. The OLED and buzzer are 5 V tolerant.

3. Installing the NanoBoy Library

Download the latest NanoBoy library ZIP from: https://github.com/ameriqbalqureshi/NanoBoy

Either download the library itself (Nanoboy.zip) or download the entire bundle. If you download entire bundle, extract it first to get the actual library, NanoBoy.zip.

- Open the **Arduino IDE**.
- Go to **Sketch → Include Library → Add .ZIP Library…**
- Select the downloaded NanoBoy.zip.
- After installation, check under **File → Examples → NanoBoy**.

You're ready to load your first program!

> The Library by default compiles for 1.3" OLED display that uses SH110X controller. If using 0.96" OLED with SSD1306 controller, select appropriate display in NanoBoy.h file in the libraries folder.

4. Selecting Your OLED Type

NanoBoy supports both SSD1306 (0.96") and SH1106 (1.3") OLEDs.

The NanoBoy library is installed in the libraries folder in the user folder or where you have pointed out your projects in the Arduino preferences. Locate the libraries folder and in it NanoBoy library folder. You will find NanoBoy.ino file that you can open with Arduino, it will open two sub-Files NanoBoy.h and NanoBoy.cpp or you can open directly NanoBoy.h file in any editor of your choice, like notepad.

At the top of this file you will see:

```
// #define NANOboy_USE_SSD1306    // for 0.96" OLED
#define NANOboy_USE_SH1106       // for 1.3" OLED
```

To use SSD1306, **uncomment** the first line and comment the second.

To use SH1106, keep the second line uncommented.

Only one line must be active at a time.

5. Predefined Constants

Use these constants instead of pin numbers:

BTN_UP, BTN_DOWN, BTN_LEFT, BTN_RIGHT, BTN_A, BTN_B

COLOR_WHITE, COLOR_BLACK

SCREEN_WIDTH (128), SCREEN_HEIGHT (64)

6. NanoBoy Library Commands

Below is the full list of available commands grouped by category.

Setup & Display Control

Prototype	Description
`void begin();`	Initialise NanoBoy and the OLED. Call in `setup`
`void clear();`	Clear the internal screen buffer (all pixels off).
`void display();`	Send the buffer contents to the OLED.

Text Functions

Prototype	Description
`void setCursor(int x, int y);`	Move the text cursor to `(x, y)`.
`void setTextSize(int size);`	Set text size multiplier (1 = normal).
`void setTextColor(int color);`	Set text color (`COLOR_WHITE` or `COLOR_BLACK`).
`void print(const char *text);`	Print C-string at cursor.
`void print(int num);`	Print integer number.
`void print(float num);`	Print floating-point number.
`void print(const String &text);`	Print Arduino `String`.
`void drawText(int x, int y, const char *text);`	Draw text directly at position.
`void drawText(int x, int y, const String &text);`	Same with `String`.

Graphics / Shapes

Prototype	Description
`void drawPixel(int x, int y, int color);`	Draw one pixel.
`void drawLine(int x0, int y0, int x1, int y1, int color=COLOR_WHITE);`	Draw a line.
`void drawRect(int x, int y, int w, int h, int color=COLOR_WHITE);`	Draw rectangle outline.
`void fillRect(int x, int y, int w, int h, int color=COLOR_WHITE);`	Draw filled rectangle.
`void drawCircle(int x, int y, int r, int color=COLOR_WHITE);`	Draw circle outline.
`void fillCircle(int x, int y, int r, int color=COLOR_WHITE);`	Draw filled circle.

Sprites & Collision

Prototype	Description
`void drawSprite(const Sprite &sprite);`	Draw a sprite bitmap.
`bool checkCollision(const Sprite &a, const Sprite &b);`	Return true if two sprites overlap.

Tile Maps

Prototype	Description
`void drawTileMap(const uint8_t *map, int rows, int cols, const uint8_t *tiles);`	Draw a background using tiles.

Input & Output

Prototype	Description
`bool buttonPressed(int btn);`	Check if button is pressed (`BTN_UP` etc.).
`void setLED(bool state);`	Turn the onboard LED on/off.
`void beep(int freq, int dur);`	Play tone with frequency (Hz) and duration
`void beep(int dur);`	Play default tone for duration (ms).

7. First Program

Here's the classic "Hello NanoBoy":

```
#include <NanoBoy.h>
NanoBoy nb;
void setup() {
  nb.begin();
  nb.clear();
  nb.setCursor(0, 0);
  nb.print("Hello NanoBoy");
  nb.display();
}

void loop() {
  // nothing yet
}
```

Upload it and your NanoBoy comes alive!

8. Support & Community

For tutorials, updates and examples visit:
Mikrotronics Pakistan – https://github.com/ameriqbalqureshi/NanoBoy

For questions, join our community forums and share your creations.