

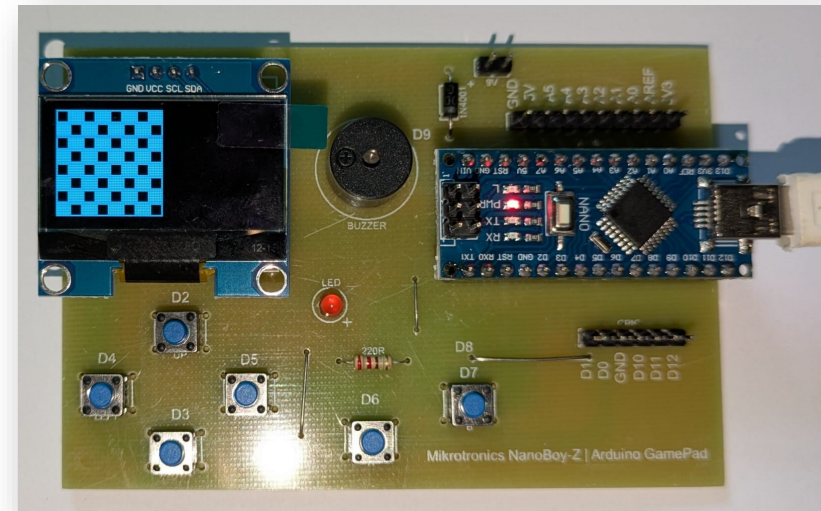
## Text

Prototype	Description
<code>void setCursor(int x, int y);</code>	Set text start position in pixels.
<code>void setTextSize(int s);</code>	Set text scaling (1=normal).
<code>void setTextColor(int c);</code>	Set text color (COLOR_WHITE or COLOR_BLACK).
<code>void print(const char* text);</code>	Print text at current cursor.
<code>void print(int num);</code>	Print integer at current cursor.
<code>void print(float num);</code>	

## Graphics Commands

Prototype	Description
<code>void drawText(int x, int y, const char *text);</code>	
	Draw text directly at specified coordinates without moving the cursor.
<code>void drawText(int x, int y, const String &amp;text);</code>	
<code>void drawPixel(int x, int y, int color);</code>	Set a single pixel on or off at (x, y).
<code>void drawLine(int x0, int y0, int x1, int y1, int color=COLOR_WHITE);</code>	Draw a straight line between two points.
<code>void drawRect(int x, int y, int w, int h, int color=COLOR_WHITE);</code>	Draw an unfilled rectangle.
<code>void fillRect(int x, int y, int w, int h, int color=COLOR_WHITE);</code>	Draw a filled rectangle.
<code>void drawCircle(int x, int y, int r, int color=COLOR_WHITE);</code>	Draw an unfilled circle with centre (x, y) and radius r.
<code>void fillCircle(int x, int y, int r, int color=COLOR_WHITE);</code>	Draw a filled circle with centre (x, y) and radius r.
<code>void drawSprite(const Sprite &amp;sprite);</code>	Draw a sprite bitmap at its x, y coordinates.
<code>void drawTileMap(const uint8_t *map, int rows, int cols, const uint8_t *tiles);</code>	Draw a tile-based background from a map array and tile set.

# NanoBoy Quick User Guide



## What is NanoBoy?

NanoBoy is a pocket-sized development board created by **Mikrotronics Pakistan** to make learning embedded systems programming fun and game-oriented.

It combines an **Arduino Nano** with a 128×64 OLED display, six buttons, an LED and a piezo buzzer — everything you need to start building your own handheld games and interactive projects without breadboards or messy wiring. Additionally other GPIO lines and Analog input lines are available as headers for interfacing with external modules if required.

The board is powered through USB connector of Arduino Nano, or if you want to run it standalone, 5V charger with USB connection can be used to power through same USB Port. An external 6-9V power header is also provided, it goes through Arduino Nano's 5V and 3.3V regulators therefore the

current draw will be limited. See Arduino Nano's current capacity. This is sufficient for this board, but if additional modules are powered through it, keep the current limits in mind.

## Hardware at a Glance

Part	Function	Arduino Nano Pin
OLED display (SSD1306/SH1106)	128×64 pixels, I <sup>2</sup> C	SDA → A4, SCL → A5
Button UP	Input, pullup	D2
Button DOWN	Input, pullup	D3
Button LEFT	Input, pullup	D4
Button RIGHT	Input, pullup	D5
Button A	action button, pullup	D6
Button B	action button, pullup	D7
LED	user LED	D8
Buzzer	sound	D9
Power	Via USB	5V
External Power	Header	6-9V

## Library Download and Install

<https://github.com/ameriqbalqureshi/nanoboy>

Download the library as .zip file and extract it. The Library is named nano-boy.zip Install it using Arduino Library manager.

It will appear as NanoBoy in libraries as well as examples.

## Library Constants

Use these in your sketches:

Constant	Meaning
BTN_UP	Up button
BTN_DOWN	Down button
BTN_LEFT	Left button
BTN_RIGHT	Right button
BTN_A	A button
BTN_B	B button
COLOR_WHITE	Pixel on
COLOR_BLACK	Pixel off
SCREEN_WIDTH	128
SCREEN_HEIGHT	64

## NanoBoy Library Commands

(Include the header at the top of your sketch: `#include <NanoBoy.h>` and create the object: `NanoBoy nb;`)

### Setup / Display Control

Prototype	Description
<code>void begin();</code>	Initialise NanoBoy hardware. Call once in <code>setup()</code> .
<code>void clear();</code>	Clear the screen buffer.
<code>void display();</code>	Transfer buffer to the OLED.

The default Display is: 1.3" SH110X display. To change to 0.96" SSD1306 Display Open the NanoBoy.ino file in the libraries and in NanoBoy.h file, uncomment the SSD1306 display and comment out the other display