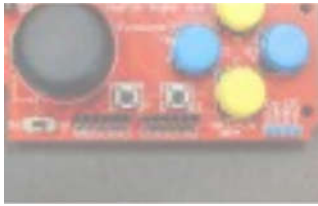


HOME » PRODUCTS » SWITCHES & USER INPUT » **FUNDUINO JOYSTICK SHIELD V1.A**

March 12, 2020

**Shop**





# Funduino Joystick Shield V1.A

\$5.95

Arduino shield provides game and robotic controller functionality.

4 in stock

1



**ADD TO CART**

Add to Wishlist

SKU: SHD-5

Categories: Arduino, MCUs & IOT, Switches & User Input

Tags: Joystick, Shield

DESCRIPTION

REVIEWS (0)

## DESCRIPTION

The Funduino Joystick Shield V1.A is an Uno compatible shield that allows you to turn an Arduino Uno or compatible into a game console or robotic controller.

## PACKAGE INCLUDES:

- Funduino Joystick Shield V1.A

## KEY FEATURES OF FUNDUINO JOYSTICK SHIELD:

- 2-Axis joystick
- 4 large buttons
- 2 small buttons
- Bluetooth / Serial interface
- I2C interface
- nRF24L01 interface

- Nokia 5110 LCD interface
- Interface connector
- Power switch to switch between 3.3 and 5V

## 2-Axis Joystick:

The X-Axis potentiometer of the joystick is connected to **A0**. The Y-Axis potentiometer is connected to **A1**. The analog inputs on a microcontroller read values over a range of 0-1023 (for typical 10-bit ADC inputs). The X-Axis and Y-Axis controls should read around 512 (midpoint) when the control is at rest. As the joystick is moved, one or both of the controls will register higher or lower values depending on how the control is being moved. The joystick also has a button 'K' which is activated by pressing the joystick down. This button is connected to **D8**.

## Buttons:

There are a total of 6 buttons on the board (not including the one on the joystick) labeled A-F. The 4 large buttons are typically used for up/down/left/right or similar functions. The two smaller buttons are typically used for less commonly used functions such as 'select' or 'start' since they are less accessible / less likely to be pressed accidentally. All buttons have pull-up resistors and pull to ground when pressed.

- Button A – Connects to **D2**
- Button B – Connects to **D3**
- Button C – Connects to **D4**
- Button D – Connects to **D5**
- Button E – Connects to **D6**
- Button F – Connects to **D7**

## Bluetooth Connector:

The **RX/TX** lines are brought out to a separate 4-pin female header along with 3.3V and Ground. This can be used for connecting a 4-pin 3.3V Bluetooth device or a TTL serial device.

## I2C Connector:

The I2C **SDA** and **SCL** lines are brought out to a separate 4-pin male header along with 5V and Ground. This is in addition to the normal **A4/A5** location of these lines. This allows for easy attachment of I2C devices.

## nRF24L01 Connector:

This connector allows an nRF24L01 RF transceiver module to be plugged in.

### 2 x 4 Female Header

- GND – Ground.
- VCC – 3.3V
- CE – Connects to **D9**
- CSN – Connects to **D10**

- SCK – Connects to **D13**
- MOSI – Connects to **D11**
- MISO – Connects to **D12**
- IRQ – No Connection

## Nokia 5110 LCD Connector:

This female header connector is designed to mount the Nokia 5110 LCD that was originally designed for Nokia phones and provides a 48×84 pixel matrix.

This interface occupies the same **D9-D13** pins as the nRF24L01, so you can't use both at the same time.

## Interface Connector:

This dual row yellow male header connector provides another point of access to all the buttons, joystick pots, 3.3V, 5V and Ground. The pin-out of this connector is labeled on the board to the left of the connector.

Here is an off-site link to a **schematic** of the assembly that you might find helpful.

## OUR EVALUATION RESULTS:

These are very convenient for easily adding game control or manual robotic control type of functionality to a project without having to worry about how to mount stuff.

The program below checks the status of the buttons and joystick every second and prints out the results to the Serial Monitor Window.

## Funduino Joystick Shield Example Program

```
/* Funduino Joystick Shield Example
```

This program simply captures all of the inputs of the Funduino Joystick Shield by joystick every second and displays them in the Serial Monitor Window.

The Arduino pins below are defined by the shield and cannot be changed.

```
*/
```

```
int const UP_BTN = 2;
```

```
int const DOWN_BTN = 4;
```

```
int const LEFT_BTN = 5;
```

```
int const RIGHT_BTN = 3;
```

```
int const E_BTN = 6;
```

```
int const F_BTN = 7;
```

```
int const JOYSTICK_BTN = 8;
```

```
int const JOYSTICK_AXIS_X = A0;
```

```
int const JOYSTICK_AXIS_Y = A1;
```

```
int buttons[] = {UP_BTN, DOWN_BTN, LEFT_BTN, RIGHT_BTN, E_BTN, F_BTN, JOYSTICK_BTN};
```

```
//=====
```

```
// Initialization
```

```
//=====
```

```
void setup() {
```

```
    //Set all button pins as inputs with internal pullup resistors enabled.
```

```
    for (int i; i < 7; i++) pinMode(buttons[i], INPUT_PULLUP);
```

```
    Serial.begin(9600);
```

```
}
```

```
//=====
```

```
// Main
```

```
//=====
```

```
void loop() {
```

```
    // Check each button input and print the status to the Serial Monitor Window
```

```
    Serial.print("UP = "), Serial.print(digitalRead(UP_BTN)), Serial.print("t");
```

```
    Serial.print("DOWN = "), Serial.print(digitalRead(DOWN_BTN)), Serial.print("t");
```

```
    Serial.print("LEFT = "), Serial.print(digitalRead(LEFT_BTN)), Serial.print("t");
```

```
    Serial.print("RIGHT = "), Serial.print(digitalRead(RIGHT_BTN)), Serial.print("t");
```

```
    Serial.print("E = "), Serial.print(digitalRead(E_BTN)), Serial.print("t");
```

```
    Serial.print("F = "), Serial.print(digitalRead(F_BTN)), Serial.print("n");
```

```
    Serial.print("JOYSTICK BTN = "), Serial.print(digitalRead(JOYSTICK_BTN)), Serial.print("t");
```

```
    // Map the X/Y joystick to basic -1/0/1 to indicate basic direction of the joystick
```

```
    Serial.print("X DIR = "), Serial.print(map(analogRead(JOYSTICK_AXIS_X), 0, 1023, -1, 0, 1));
```

```
    Serial.print("Y DIR = "), Serial.print(map(analogRead(JOYSTICK_AXIS_Y), 0, 1023, -1, 0, 1));
```

```
    // Print the full X/Y joystick values (0-1023)
```

```
    Serial.print("X = "), Serial.print(analogRead(JOYSTICK_AXIS_X)); Serial.print("t");
```

```
    Serial.print("Y = "), Serial.print(analogRead(JOYSTICK_AXIS_Y)); Serial.println("t");
```

```
    delay(1000);
```

```
}
```

**BEFORE THEY ARE SHIPPED, THESE MODULES ARE:**