



# SparkFun Blocks for Intel® Edison - GPIO Block

## Introduction and Overview

The General Purpose Input/Output (GPIO) Block breaks out the simple GPIO functionality of the Intel Edison. Using selectable level shifters, it is possible to use VSYS or 3.3v logic levels with this Block.



*GPIO Block*

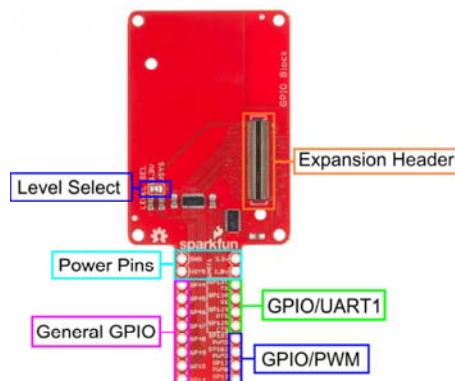
## Suggested Reading

If you are unfamiliar with Blocks, take a look at the General Guide to Sparkfun Blocks for Intel Edison.

Other tutorials that may help you on your Edison adventure include:

- Edison Getting Started Guide
- Logic Levels

## Board Overview



### *GPIO Block Functional Diagram*

- Level Select - Jumper selects GPIO reference voltage, Can be set to 3.3v or VSYS
- Power Pins - Raw access to power pins on Edison
  - GND - Ground pin for all blocks and Edison
  - VSYS - Raw input for Edison and all Blocks.
    - Normal output (with power blocks) 4.0V-4.1V.
    - You can power an Edison through this pin. Acceptable voltages 3.3V-4.5V
  - 1.8v - 1.8v supplied by Edison internal power supply
  - 3.3v - 3.3v supplied by Edison internal power supply
- General GPIO - General use GPIO pins.
- GPIO/UART1 - GPIO pins that can also be used as a second UART. (Useful for GPS receivers and other serial devices)
- GPIO/PWM - GPIO pins capable of generating PWM waveforms. (Useful for LED dimming and Motor control)
- Expansion Header - The 70-pin Expansion header breaks out the functionality of the Intel Edison. This header also passes signals and power throughout the stack. These function much like an Arduino Shield.

## Using the GPIO Block

To use the GPIO Block, attach an Intel® Edison to the back of the board, or add it to your current stack. Blocks can be stacked without hardware, but it leaves the expansion connectors unprotected from mechanical stress.



*GPIO Block Installed*

We have a nice Hardware Pack available that gives enough hardware to secure three blocks and an Edison.

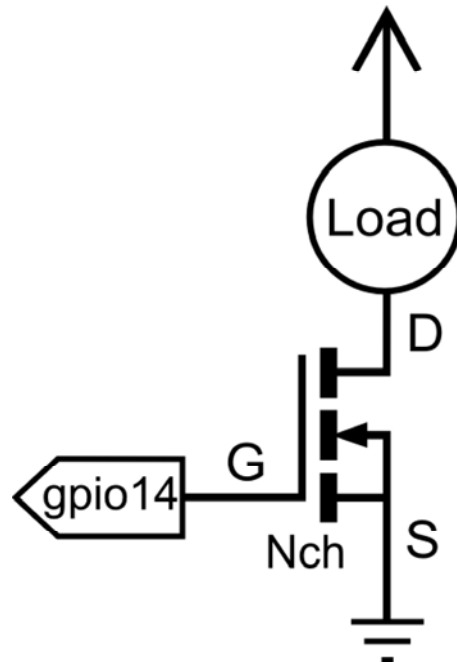


#### *Intel Edison Hardware Pack*

NOTE: The GPIO Block does not have console access or a voltage regulator. It is recommended to use a console communication block in conjunction with this block like ones found in the General Guide to Sparkfun Blocks for Intel Edison.

### Using the GPIO Block as an output device

If you want to use the GPIO Block to control high power LED's or Relays, an external transistor or MOSFET will be required. It is possible to illuminate a small LED directly from the level shifter. It may not be as bright since the current output of the TXB0108 level converter is very low (~5ma).



*Connection Diagram for Load (LED, Motor, or Relay)*

In the terminal we will demonstrate how to activate and use a GPIO pin as an output.

First navigate to the GPIO directory on the Edison.

```
cd /sys/class/gpio
```

Select the GPIO pin to enable. In this case, we used GPIO 14.

```
echo 14 > export
```

Navigate to the newly created GPIO directory.

```
cd gpio14
```

If you type "ls", you should see a bunch of variables.

```
active_low  direction  power      uevent
device      edge        subsystem  value
```

Let's set the "direction" of the port to output

```
echo out > direction
```

To confirm this, we will "cat" the value

```
cat direction
```

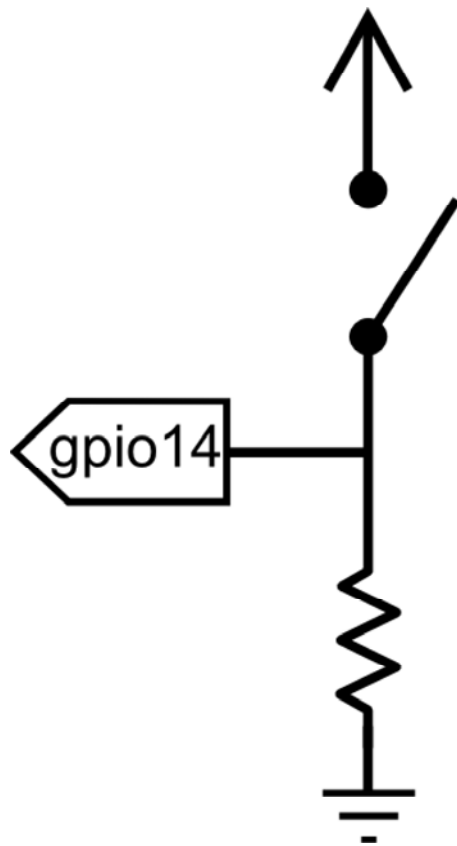
You should see the "out" in the command line. Now the device is configured as an output. "value" is where the status of the pin is set, 1 for high, 0 for low.

```
echo 1 > value
```

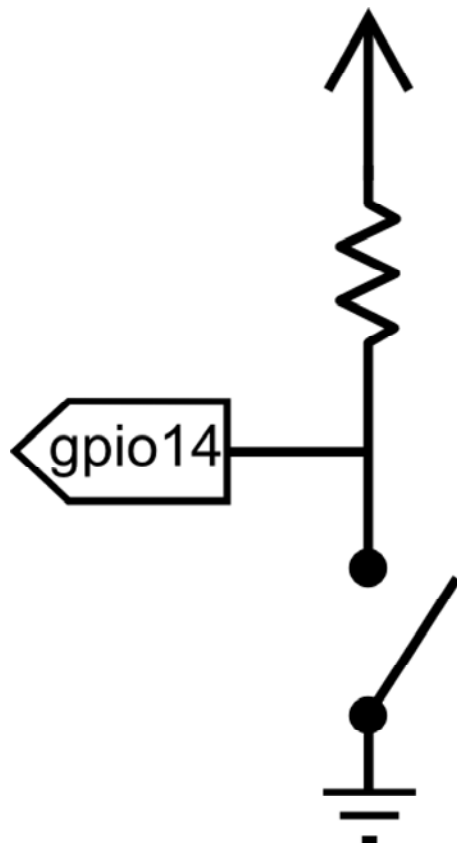
Testing with a multi-meter, small led, or oscilloscope, you should see a "high" status (3.3V) present on gpio14.

## Using the GPIO Block as an input device.

If you want the GPIO Block to read switches, buttons, or other logic level inputs, you must pay attention to pull-up and pull-down resistors. The level converter on board is very weak. Here are two scenarios explained:



*Connection Diagram for Active High Push Button*



*Connection Diagram for Active Low Push Button*

In the terminal, we will demonstrate how to activate and use a GPIO pin as

an input configured as an active high.

First, navigate to the GPIO directory on the Edison.

```
cd /sys/class/gpio
```

Select the GPIO pin to enable. In this case let us use GPIO 14.

```
echo 14 > export
```

Navigate to the newly created GPIO directory.

```
cd gpio14
```

If you type "ls", you should see a bunch of variables.

```
active_low  direction  power      uevent
device      edge       subsystem  value
```

Let's set the "direction" of the port to output.

```
echo in > direction
```

To confirm this, we will "cat" the value.

```
cat direction
```

You should see the "in" in the command line. Now the device is configured as an input. "value" is where the status of the pin is set, 1 for high, 0 for low.

```
cat value
```

With a button pressed, you should see a 1. When the button is not pressed you should see a 0. Using the up arrow, you can recall previously run commands.

## Resources and Going Further

Now that you have had a brief overview of the GPIO Block, take a look at some of these other tutorials. These tutorials cover programming, Block stacking, and interfacing with the Intel Edison ecosystems.

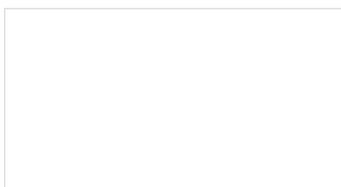
### Edison General Topics:

- General Guide to Sparkfun Blocks for Intel Edison
- Edison Getting Started Guide - Programming with Arduino
- Loading Debian (Ubinix) on the Edison

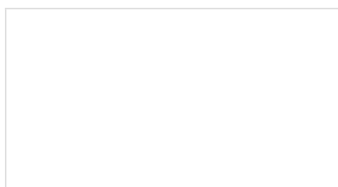
### Block Specific Topics:

- Logic Levels
- GPIO Block Github repo

Check out these other Edison related tutorials from SparkFun:



SparkFun Blocks for Intel®



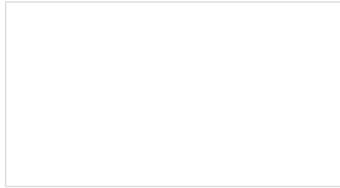
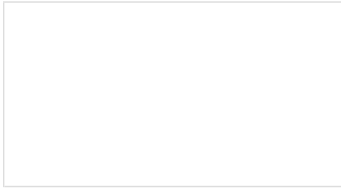
Edison Getting Started

### Edison - Battery Block

A quick overview of the features of the Battery Block.

### Guide

An introduction to the Intel® Edison. Then a quick walk through on interacting with the console, connecting to WiFi, and doing...stuff.



### Loading Debian (Ubilinux) on the Edison

How to load a Debian distribution (specifically Ubilinux) onto the Edison.

### Installing libmraa on Ubilinux for Edison

libmraa is a tool kit for interacting with various Intel single board computers.