

Note: This presentation was made and provided by Intel during the Intel Embedded Education & Research Summit in March 2015

Intel Edison Workshop

Setting up Edison Step by Step

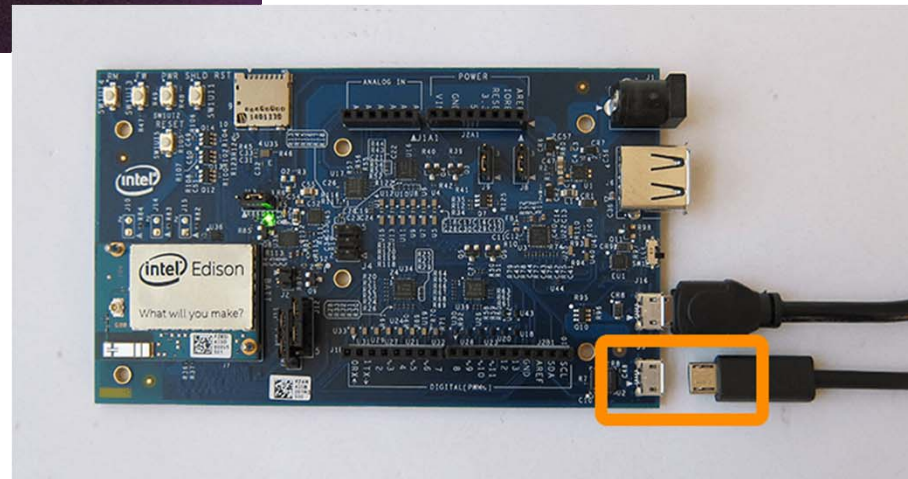


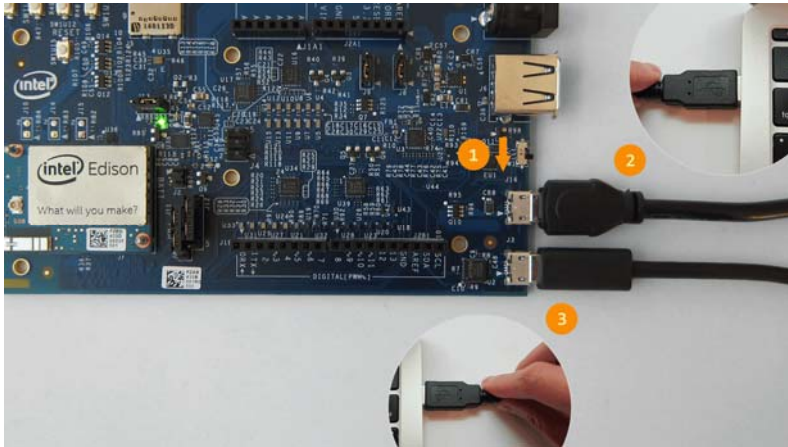
Our Workshop Goal:

1. Unbox Edison
2. Learn how to connect and configure Edison board:
Serial connecton
Name /WiFi set up/Password
3. Install drivers (New Windows Installer amd manual install)
4. Intel Development IoT Kit
5. Install IDEs
6. Run example code

Intel® Edison Arduino Expansion Board Assembly

<https://software.intel.com/en-us/articles/intel-edison-arduino-expansion-board-assembly>





Microswitch and USB Ports Details

The slider switches between USB host mode and USB device mode.

Device mode: The switch is toggled down and a micro-USB cable can be used to turn the Intel® Edison into a computer peripheral. Device mode allows you to do such things as: program the board over USB, or mount the onboard flash memory like a disk drive.

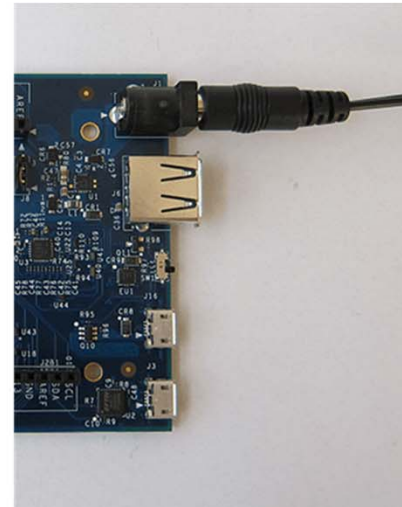
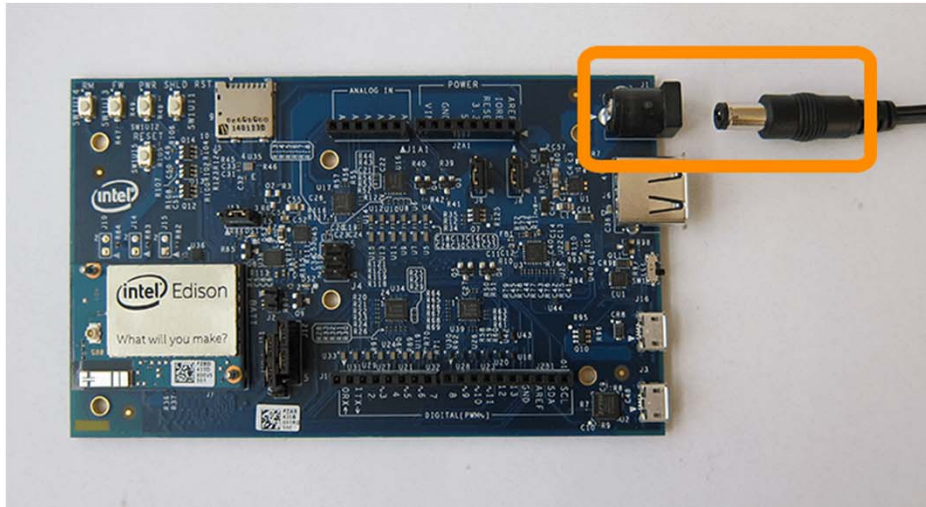
Host mode: The switch is toggled up and USB peripherals with a standard-sized USB cable (such as mice, keyboards, etc) can be plugged into the Intel® Edison. USB host mode requires the use of an external power adapter.

The Intel Edison board has three USB ports:

The **middle port** (Micro A type) is used for the following:

- Power through USB
- Ethernet over USB
- Uploading Arduino sketches
- Updating the firmware by using the board as a storage device, like a flash drive

The **edge port** (Micro A type) is used to create a terminal connection by serial over USB only.



Power Through DC Plug

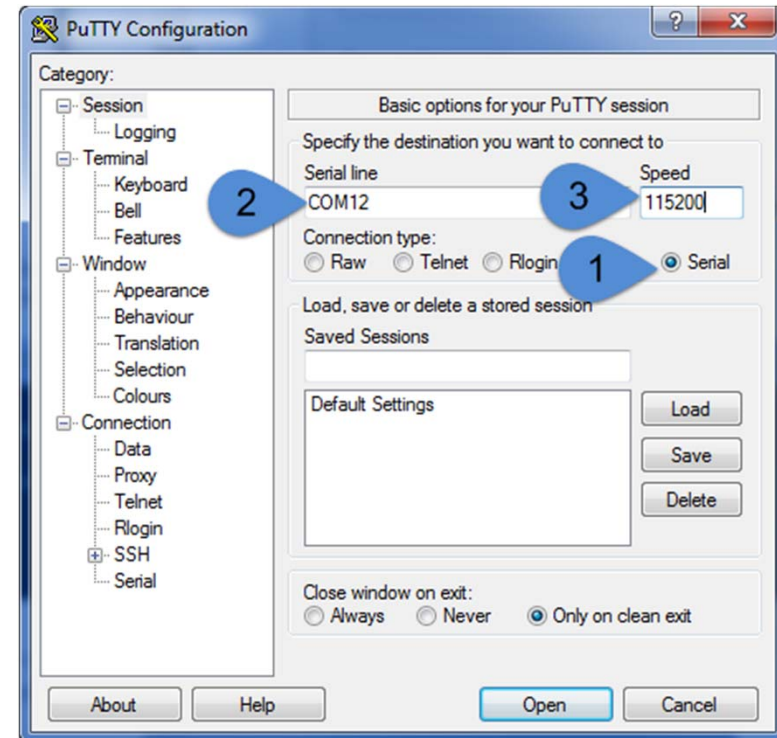
If you are going to use more power intensive features such as Wi-Fi, a servo motor, or an Arduino shield, use a DC power supply in addition to the device mode micro-USB cable.

1. Plug in a 7 to 15 VDC input DC power supply to the barrel connector. (This DC power supply could be plugged into the wall, or be part of a battery pack.)

Use Device manager to find out what COM port used by Edison serial



Putty configuration



```
COM8 - PuTTY  
Poky (Yocto Project Reference Distro) 1.6 Edison1 ttyMFD2  
Edison1 login: █
```

At the login prompt, type

```
1 █ root
```

```
COM8 - PuTTY  
root@Edison1:~# █
```

```
# configure_edison --help
```

```
COM6 - PuTTY

Poky (Yocto Project Reference Distro) 1.6.1 edison-3 ttyMFD2

edison-3 login: P@ssw0rd
Password:
Login incorrect

edison-3 login: root
Password:
root@edison-3:~# configure_edison --help
usage: configure_edison [-h] [--server]
                        [[--setup | --name | --password | --wifi]
                        [--showWiFiIP | --version | --latest-version | --upgrade
                        | --disableOneTimeSetup | --enableOneTimeSetup | --flash <version> [<release na
me> ...]
                        | --flashFile <image-file> | --showNames]

optional arguments:
  -h, --help            show this help message and exit
  --server              Starts the server (testing only)
  --setup              Goes through changing the device name, password, and
                        wifi options
  --name               Changes the device name
  --password           Changes the device password
  --wifi              Changes the wifi options
  --showWiFiIP         IP address associated with the wireless interface
  --version            Gets the current firmware version
  --latest-version     Gets the latest firmware version
  --upgrade            Downloads the latest firmware
  --disableOneTimeSetup Disable one-time setup and WiFi access point
  --enableOneTimeSetup Enable one-time setup and WiFi access point
  --flash <version> [<release name> ...] Downloads and flashes an image
  --flashFile <image-file> Flashes the given image (.zip).
  --showNames          Show device name and SSID
root@edison-3:~#
```



```
# configure_edison --setup
```

IoT Development Kit officially released 3/5/2015

The new release offers a number of enhancements, bug fixes and improved usability including:

1. Improved Eclipse C/C++ new project creation, templates and samples to get started quickly
2. Support for 70+ sensors via UPM libraries
3. Guided Windows* installer for Intel® Edison setup and configuration
4. IO library support for Intel Edison mini breakout boards
5. Improved documentation including a Developer Resources page for Intel® Edison and guides for getting started with Wi-Fi* and Bluetooth
6. Wyliodrin* (visual programming) support for Intel Edison

Intel® IoT Developer Kit v1.0

Hardware components

+

Software image

+

Support for various IDEs

+

Cloud services

+

Additional tools and solutions

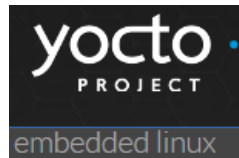
=



+



=



+

helpful Linux tools (GCC tool chain, perf, oProfile, etc.), required drivers (WiFi, Bluetooth, etc.), useful API libraries, and daemons like Lighttpd and Node.js.

=



For C/C++

+



For java, node.js, html5

+



For Arduino sketches

+



For Visual Programming

=



Intel IoT Analytics includes capabilities for data collection, storage, visualization, and analysis of sensor data. Free service.

=

[Intel System Studio for IoT](#)

+

[Wind River VxWorks* for Makers](#)

IoT Target audience and developer path

Arduino Developer	Visual Programming	JavaScript Developer	C / C++ Developer
Yocto Linux SPI	Yocto Linux	Yocto Linux	Yocto Linux, Windows, VxWorks, WR Linux
Arduino* IDE Win / Mac/ Linux*	Wylodrin Web	XDK Win/ Mac/ Linux	Eclipse Win/ Mac / Linux
Arduino Sketch C++	Visual Python	JavaScript (Node JS)	C/ C++
Quark (Galileo, Edison)	Quark (Galileo, Edison)	Quark (Galileo, Edison)	Quark, Atom, Core, Xeon
Arduino Libraries	Wylodrin*	Intel® XDK IoT Edition	Intel® System Studio Wind River* Workbench

Arduino impact in maker space

• Before

- Learn, buy, solder together components (caps, controllers, chips, regulators etc)
- Learn complex toolchains, buy specialized hardware

• After

- Open sourced HW & SW
- Huge online community & ecosystem
- Radically simplified development

Evaluation kits



Atmel EVK1105AU (ATEVK1105AU)

SKU: ATEVK1105AU

UC3A0512AU evaluation kit

✓ In store: 10+

\$179

Quantity: 1 [Add to Cart](#)



```
File Edit View Terminal Tabs Help
GNU nano 2.0.7 File: t-arm-elf Modified
LIBIASMRC = arm/libifuncs.asm
LIBIASMFUNCS = udivsi3 divsi3 umodsi3 modsi3 dvmtd_tls_bb_init_func \
    call_via_rx_interwork call_via_rx \
    lshrdi3 ashrdi3 ashldi3 \
    negdf2 addsubdf3 muldivdf3 cmpdf2 unorddf2 fixdfsi fixunsdfsi \
    truncdfsf2 negsf2 addsubsf3 muldivsf3 cmprsf2 unordsf2 \
    fixsfsi fixunssf3 floatdidf floatdisf floatundidf floatundisf

MULTILIB_OPTIONS = marm/thumb mno-thumb-interwork/thumb-interwork
MULTILIB_DIRNAMES = arm/thumb normal interwork
MULTILIB_EXCEPTIONS =
MULTILIB_MATCHES =

#MULTILIB_OPTIONS += marm/armv7
#MULTILIB_DIRNAMES += thumb2
#MULTILIB_EXCEPTIONS += marm/armv7* marm/marmv7*
#MULTILIB_MATCHES += marm/armv7=marmv7-a
#MULTILIB_MATCHES += marm/armv7=marmv7-r
#MULTILIB_MATCHES += marm/armv7=marmv7-m

Get Help WriteOut Read File Prev Page Cut Text Cur Pos
Exit Justify Where Is Next Page UnCut Text To Spell
```



Computing Renaissance

Wylidrin* Visual Programming

(for Edison and Galileo)

Select your peripheral/sensor

Simply drag and drop 'blocks'

Tool automatically creates Javascript code

The screenshot displays the Wylidrin web interface for a project named 'leds'. The left sidebar contains a menu with categories like Program, Social, Peripherals, Grove, Play, Embedded, Internet, Web Server, Graphics, Robots, Signals, Multimedia, Boards Specialized, Older, and Bitcoins. The main workspace is divided into two panels. The top panel shows a visual programming block 'set lights to' connected to a 'create list with' block, which is then connected to a 'set music to' block. The 'set music to' block is connected to a 'Load audio stream from address' block, which is then connected to a 'Play audio stream' block. The 'Play audio stream' block is connected to a 'repeat (while)' block, which contains a 'delay 10 milliseconds' block and a 'set index to' block. The 'set index to' block is connected to a 'Stream level' block, which is then connected to a 'Set basic color' block. The 'Set basic color' block is connected to a 'lights' block. The bottom panel shows the generated JavaScript code, which includes imports for pybass and ctypes, and a series of function definitions for BASS_Init, BASS_PluginLoad, HIWORD, LOWORD, StreamLevel, colorToRGB, and basic_color.

```
from pybass import *
from ctypes import *

import struct

lights = None
music = None
index = None

BASS_Init(-1, 44800, 0, 0, 0);
BASS_PluginLoad('/usr/local/lib/libbassflac.so', 0)
BASS_PluginLoad('/usr/local/lib/libbass-aac.so', 0)

try:
    from wylidrin import *
except:
    from wiringpi2 import *
    wiringPiSetup()

def HIWORD(words):
    return words & 0x0000ffff

def LOWORD(words):
    return words >> 16

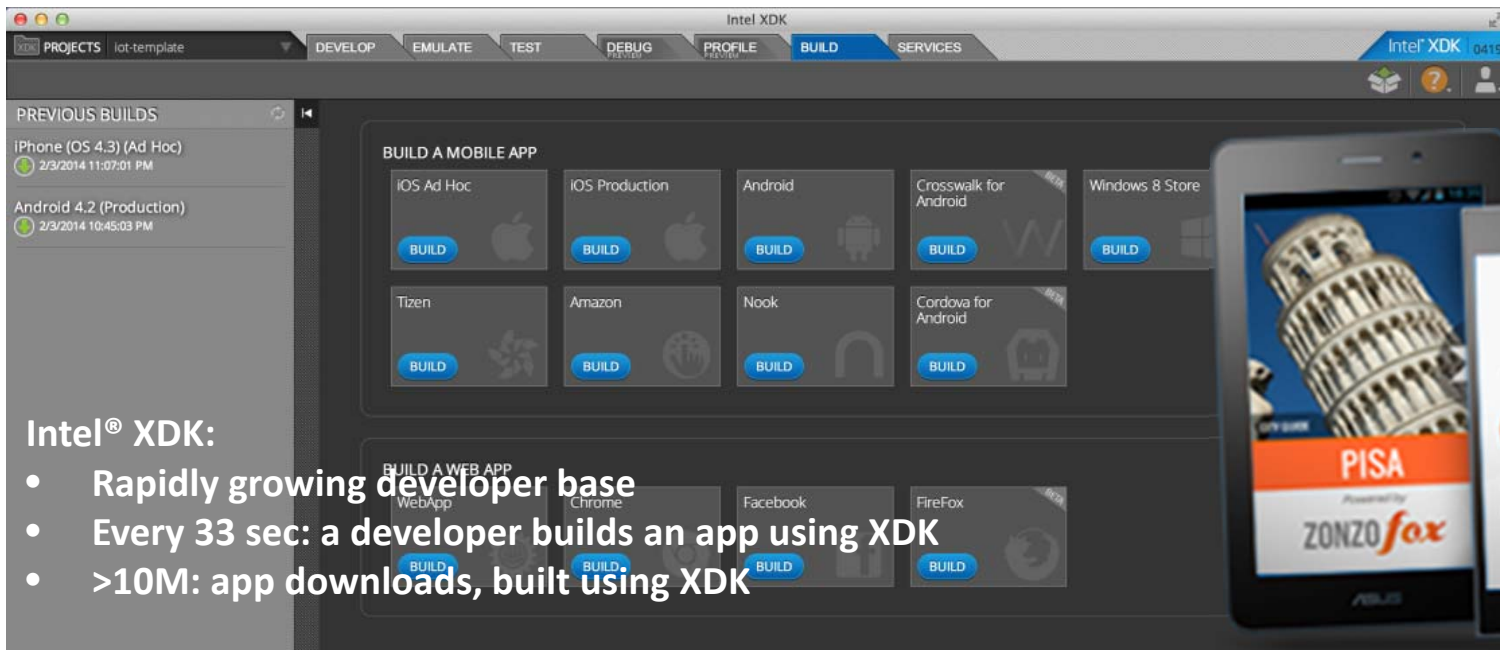
def StreamLevel(stream, scale):
    level = BASS_ChannelGetLevel(stream)
    return ((HIWORD(level)+LOWORD(level))/2)*scale/32768

def colorToRGB(color):
    return struct.unpack('B88', color[1:].decode('hex'))

def basic_color(color):
    value = 0
    if color>=128:
        value = 1
```

www.wylidrin.com/galileo

Intel® XDK

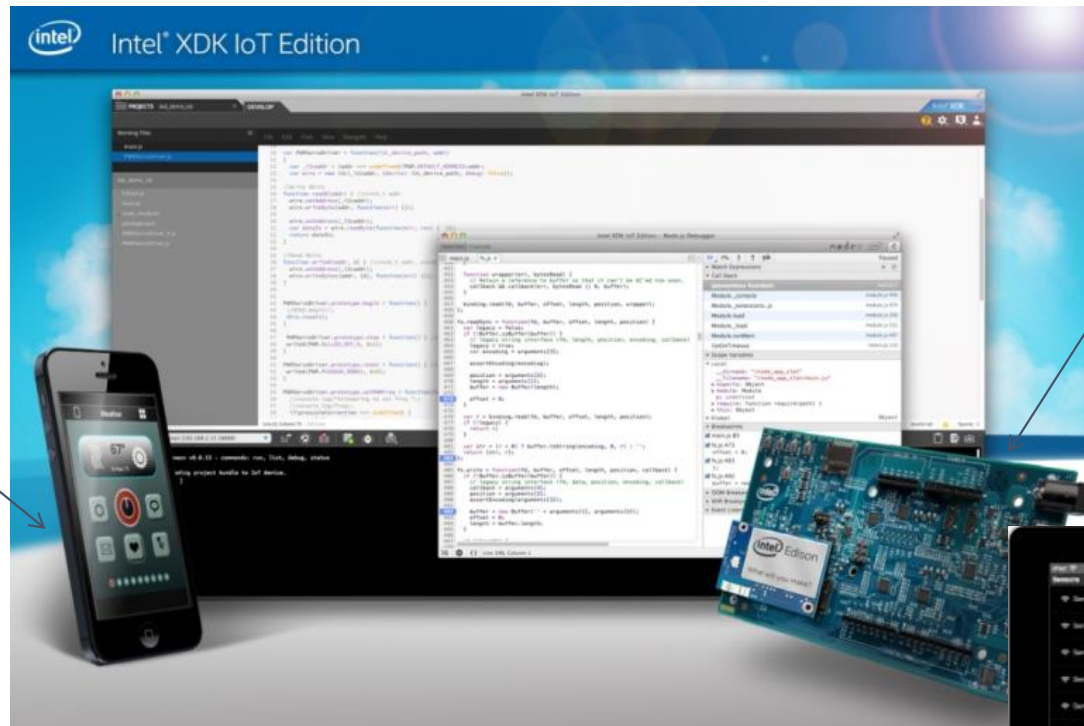


Intel® XDK:

- Rapidly growing developer base
- Every 33 sec: a developer builds an app using XDK
- >10M: app downloads, built using XDK

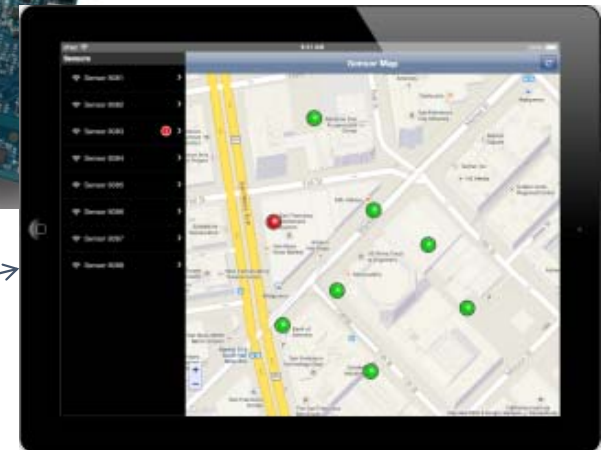
Download: <https://software.intel.com/en-us/html5/xdk-iot>

**Design companion app UI
in HTML5 for the IoT
device. Control via
phone/tablet/browser**



**Program the device
(Edison or Galileo) using
Node.JS**

**Visualize & analyze data
generated from a system of
IoT devices**



Intel® System Studio

Deliver next-gen IoT systems and applications

- Accelerate time to market
- Strengthen System Reliability
- Boost power efficiency and performance
- Software tools to Build, Debug and Tune IoT systems



intel
Software

What
thing will
you make
smart?

GET
STARTED▶

Intel® System Studio: <http://intel.ly/system-studio>



IoT Development Kit officially released 3/5/2015

Download Software for the Intel® IoT Developer Kit v1.0

Intel® Edison Updates

The dev kit libraries are already available in the operating system distributed with the Intel® Edison development board. If you already followed the latest [Intel® Edison Getting Started guide](#), you already have the latest versions.

A Windows* 64-bit installation program is now available for the Intel Edison development board.

Download the installation program

Windows* 64-bit (21MB) >



Intel® Galileo Board (Linux Boot Image)

Yocto 1.6 embedded Linux-based OS and our API libraries, useful daemons like Lighttpd and Node.js.

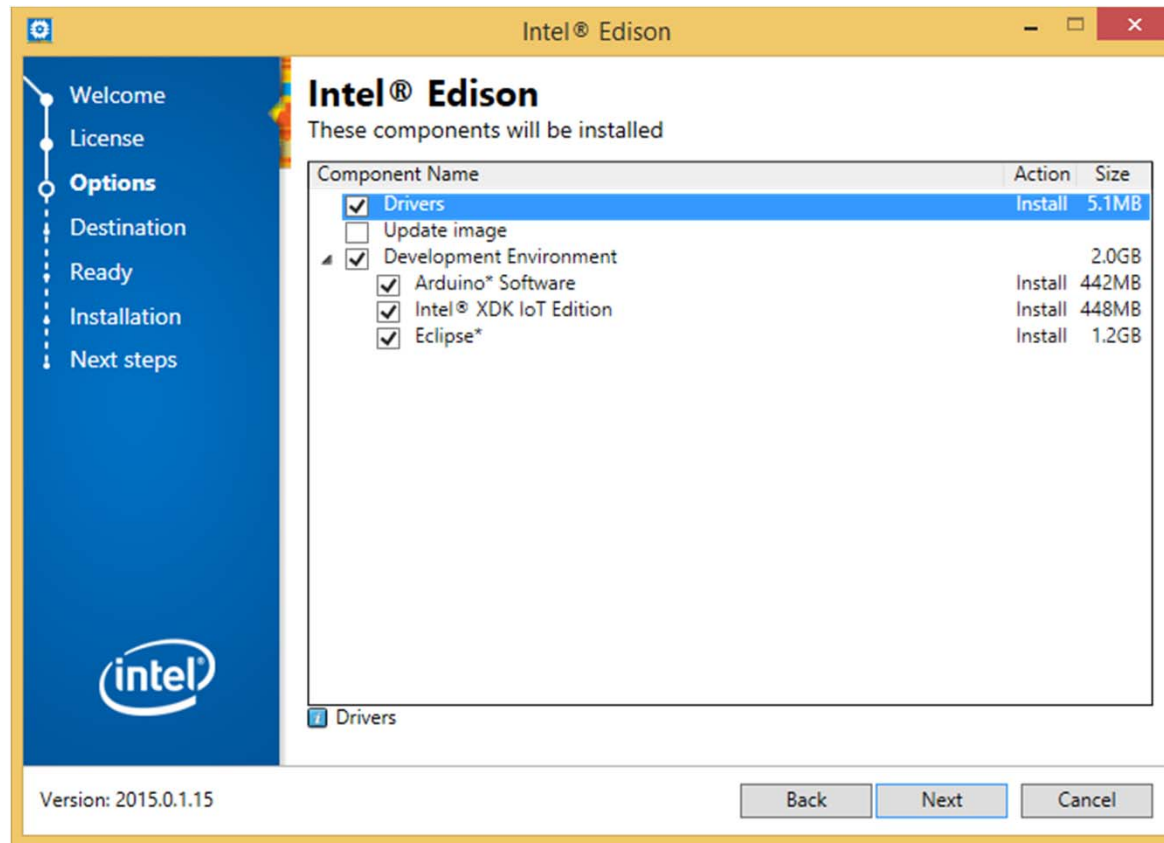
Note: Image must be copied to an SD card and is required for using development environments.

Instructions for: [Windows*](#), [Mac OS*](#) and [Linux*](#)

Download (200MB) >



IoT Development Kit officially released 3/5/2015



Flashing Edison (step-by-step):

Important note: if you Edison is brand new from factory or you need to reflash image due to corruption use this procedure

1. Unbox, connect Edison to Arduino break out board
2. Connect board with USB to PC, power-on board
3. Serial connect to board
4. Format Edison drive to FAT32 (full format) from PC
5. Copy latest Yokto image to Edison drive
(<http://www.intel.com/support/edison/sb/CS-035180.htm>)
6. In serial console: `# reboot ota`
7. After reboot verify that image was flashed `# configure_edison --version` (should be 120 or better)
8. In serial console : `#opkg update`
`#opkg upgrade`

Useful links

Setting up Edison HW

<https://software.intel.com/en-us/iot/getting-started>

<https://software.intel.com/en-us/iot/getting-started>

Starter Kit <https://software.intel.com/en-us/iot/devkit>

IDE (software) download <https://software.intel.com/en-us/iot/downloads>

Other Useful Links:

Developer Zone IoT <https://software.intel.com/en-us/iot>

Makers forum <https://communities.intel.com/community/makers>

Edison Documents and guides: <http://www.intel.com/support/maker/edison.htm#documents>

Latest Edison images, drivers, installers : <http://www.intel.com/support/edison/sb/CS-035180.htm>

Edison Arduino Breakout board documentation : http://download.intel.com/support/edison/sb/edisonarduino_hg_331191007.pdf

Edison mini breakout board documentation: http://download.intel.com/support/edison/sb/edisonbreakout_hg_331190006.pdf

Sparkfun Edison Blocks <https://www.sparkfun.com/categories/272>

Debian for Edison and Galileo (ubilinux) <http://www.emutexlabs.com/ubilinux>

