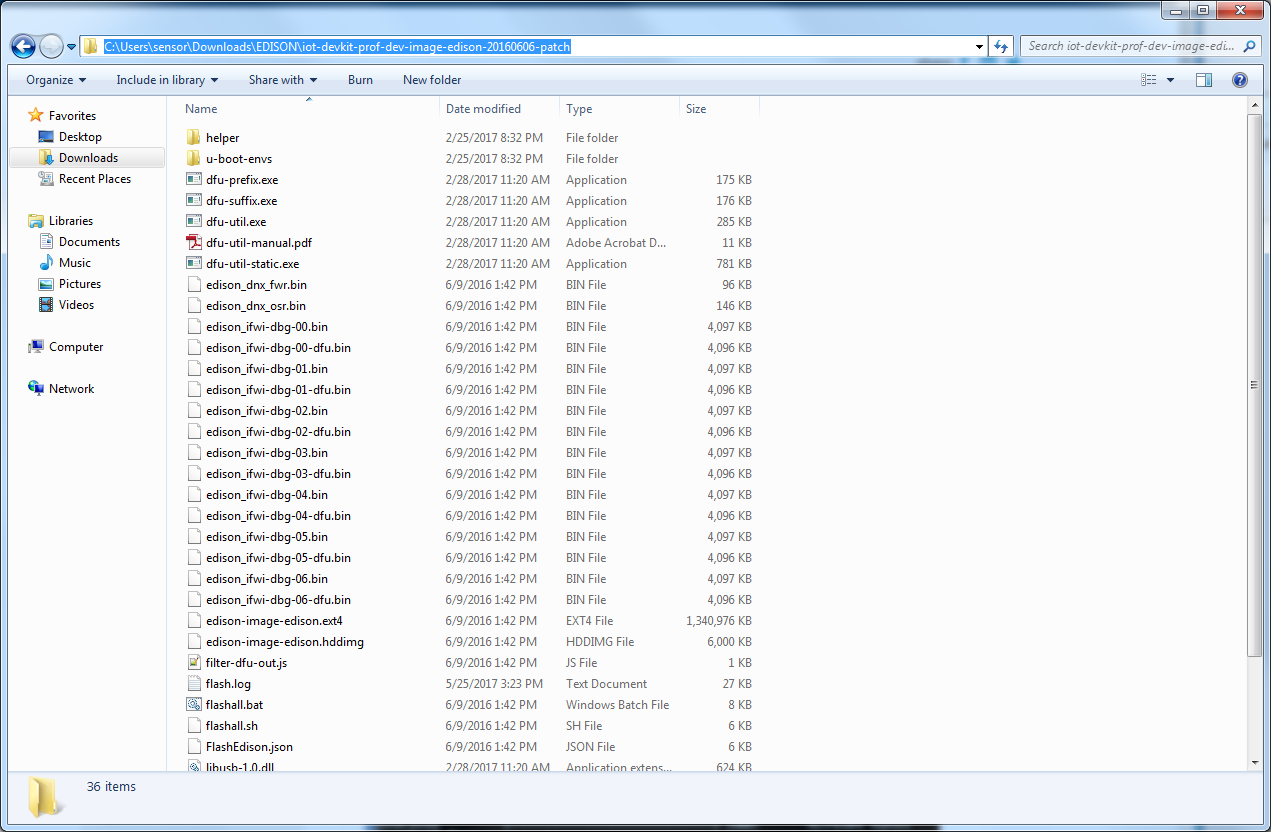
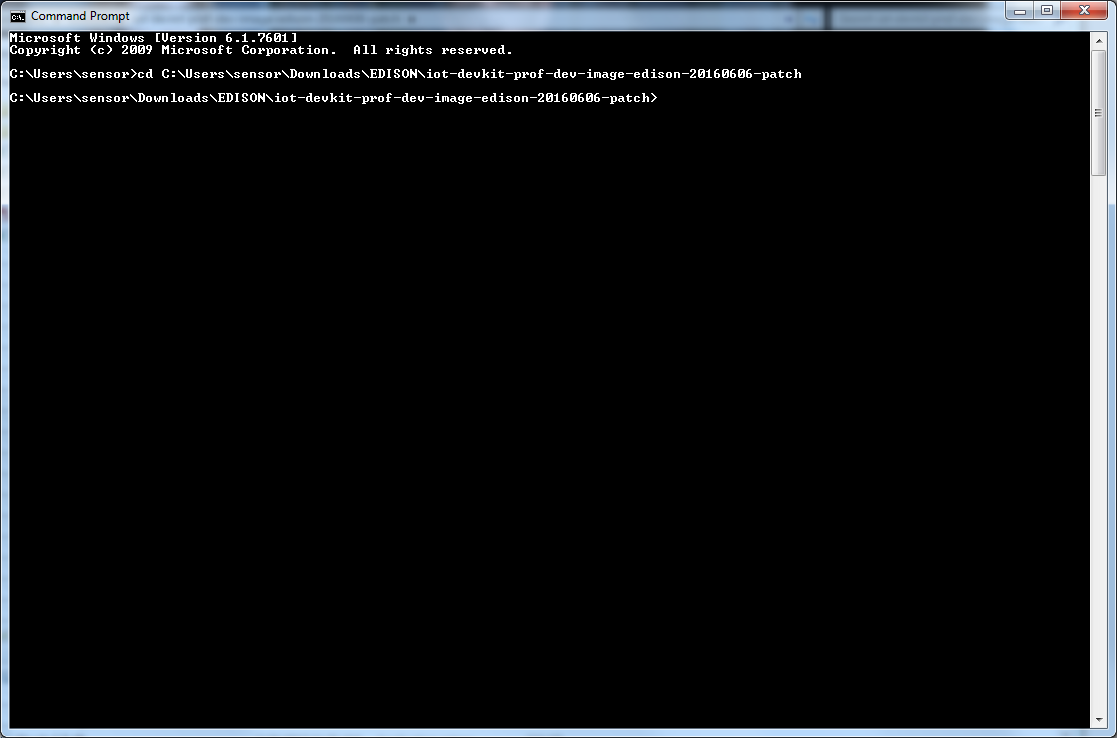
How to setup Sparkfun Intel ® Edison nodes

Flashing the firmware (Windows):

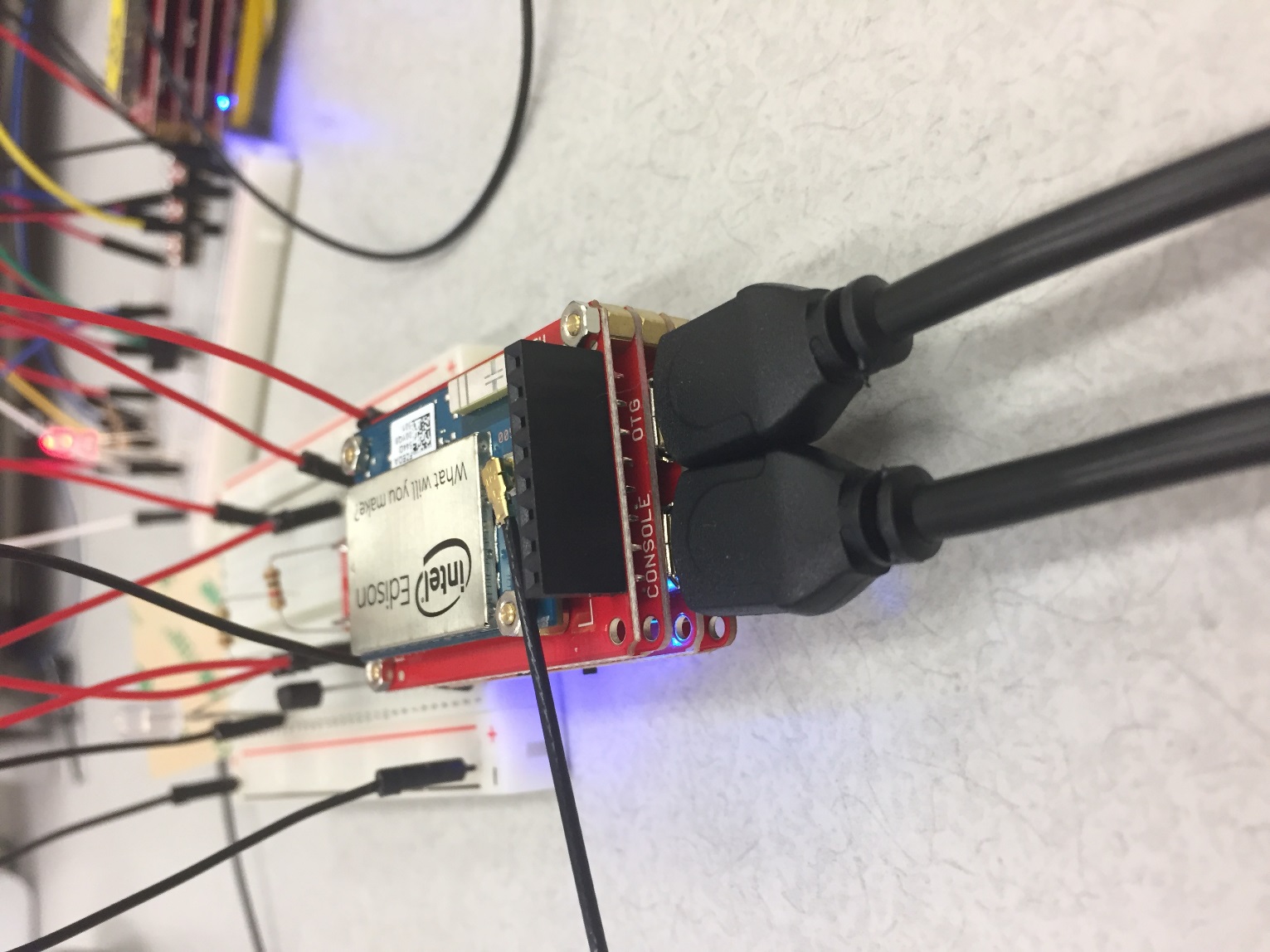
1. [Download](https://software.intel.com/iot/hardware/edison/downloads) Yocto image for the board.
2. Extract it to a folder where you want to setup everything.
3. Download dfu-util.exe and libusb-1.0.dll from <http://dfu-util.sourceforge.net/releases/> . Make sure these are placed in the same folder that you made in step 2.
4. Navigate to the folder and copy the location from the top of the file navigator.



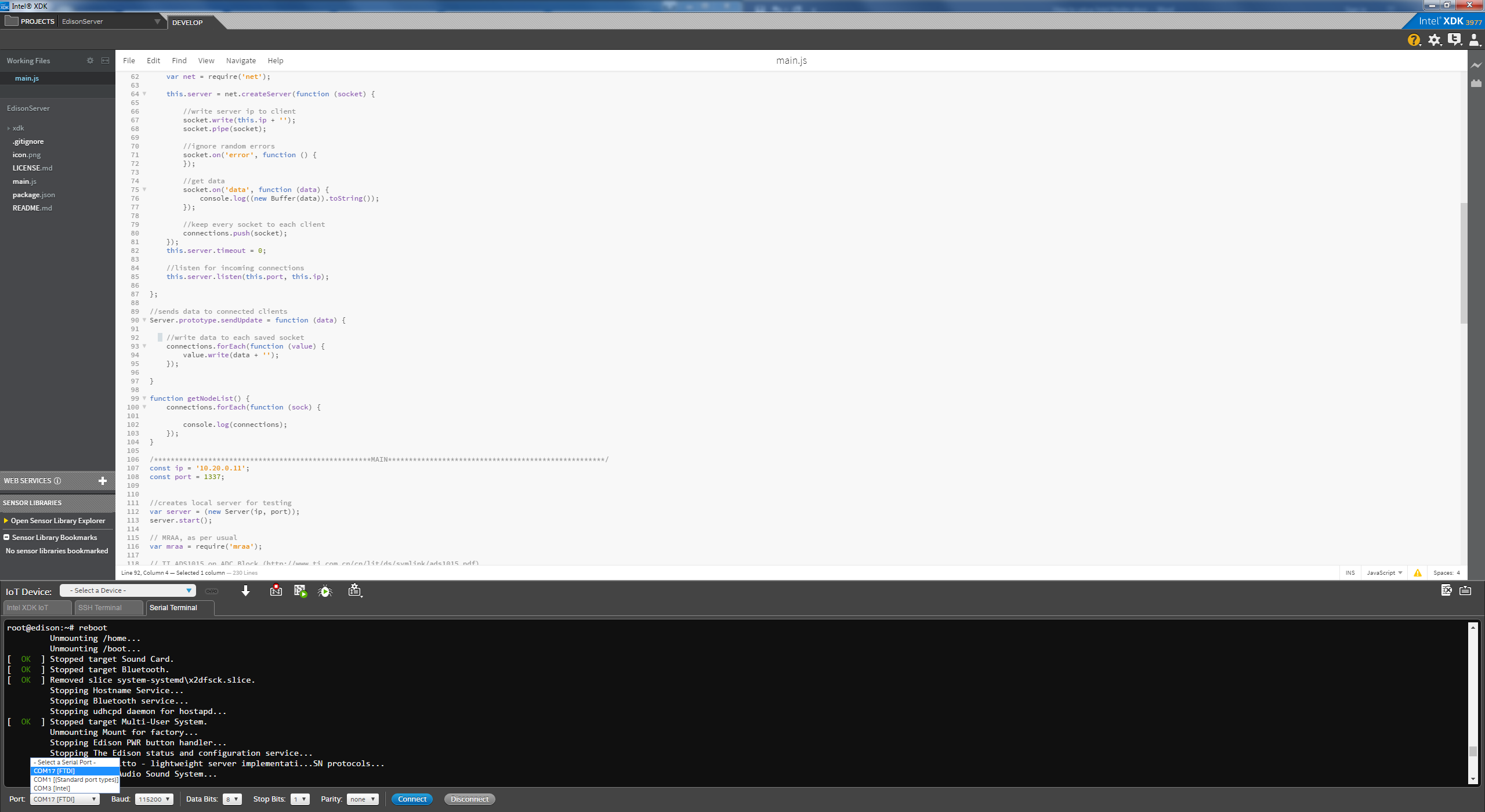
1. Open the Windows command prompt and type “cd” and paste the location you copied and hit enter.

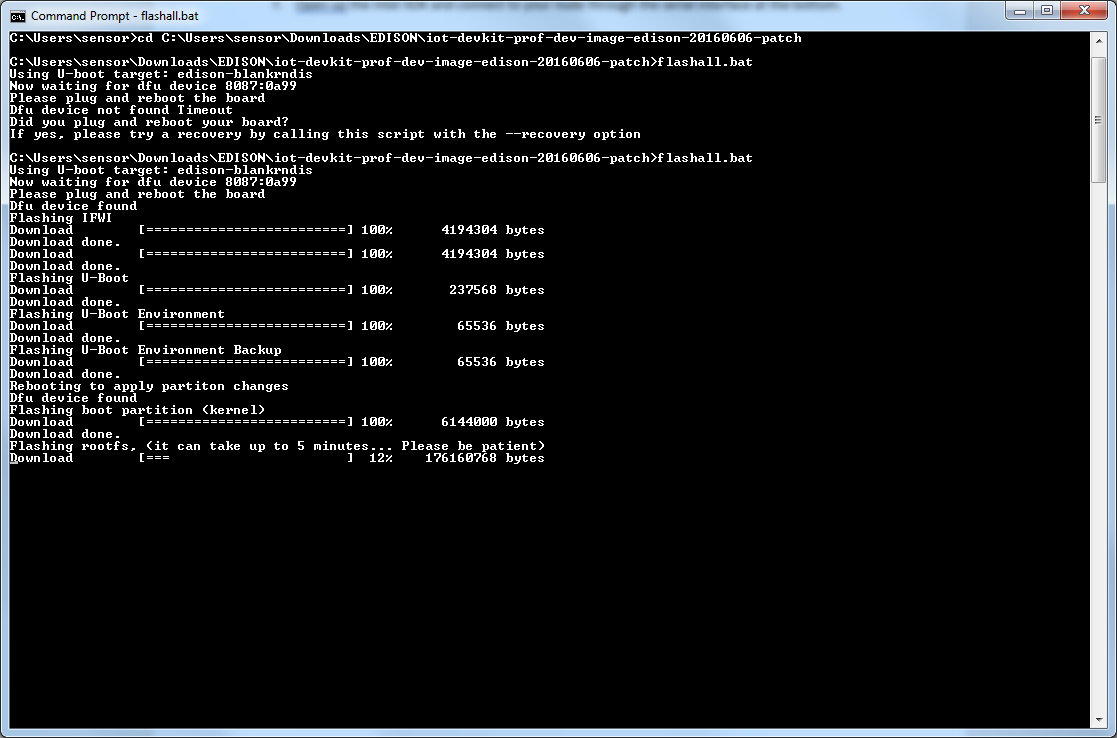
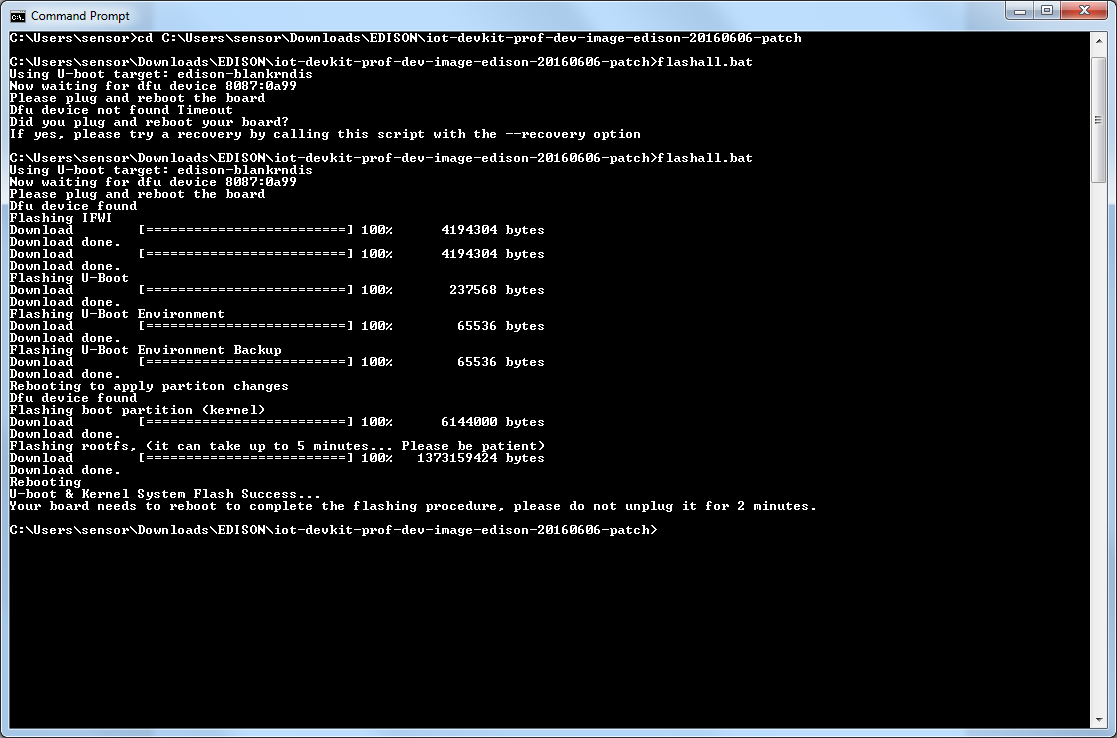


1. Connect both Micro-USB connectors to the “Console” and “OTG” ports on the Edison Stack.



1. Open the Intel XDK and connect to your node through the serial interface at the bottom.



1. Go back to the command prompt and type “flashall.bat” and hit enter. You should see something like this.
2. Go back to the Intel XDK and type “reboot” and now the command prompt should look like this.

The process should take around 5-10 minutes. DO NOT UNPLUG THE DEVICE!!!

Setup Node/Network

configure\_edison --setup

//enter password

Give this Edison a unique name.

This will be used for the access point SSID and mDNS address.

Make it at least five characters long (leave empty to skip): NAME

Is edison04 correct? [Y or N]: y

Do you want to set up wifi? [Y or N]: n

Done.

//disable power management

root@edison:~# cd /lib/systemd/system

root@edison:/lib/systemd/system# nano wifi-power-management-off.service

//paste this into the editor

[Unit]

Description=Disable power management for wlan0

Requires=sys-subsystem-net-devices-wlan0.device

After=sys-subsystem-net-devices-wlan0.device

[Service]

Type=oneshot

ExecStart=/sbin/iwconfig wlan0 power off

*//modify wpa\_client\_actions.sh*

root@edison:/lib/systemd/system# cd /etc/wpa\_supplicant/

root@edison:/etc/wpa\_supplicant# nano wpa\_cli-actions.sh

#

# This script file is passed as parameter to wpa\_cli, started as a daemon,

# so that the wpa\_supplicant events are sent to this script

# and actions executed, like :

# - start DHCP client when STA is connected.

# - stop DHCP client when STA is disconnected.

# - start DHCP client when P2P-GC is connected.

# - stop DHCP server when P2P-GO is disconnected.

#

# This script skips events if connmand (connman.service) is started

# Indeed, it is considered that the Wifi connection is managed through

# connmand and not wpa\_cli

#

IFNAME=$1

CMD=$2

kill\_daemon() {

NAME=$1

PF=$2

if [ ! -r $PF ]; then

return

fi

PID=`cat $PF`

if [ $PID -gt 0 ]; then

if ps | grep $NAME | grep $PID; then

kill $PID

fi

fi

if [ -r $PF ]; then

# file can be removed by the deamon when killed

rm $PF

fi

}

echo "event $CMD received from wpa\_supplicant"

# if Connman is started, ignore wpa\_supplicant

# STA connection event because the DHCP connection

# is triggerd by Connman

if [ `systemctl is-active connman` == "active" ] ; then

if [ "$CMD" = "CONNECTED" ] || [ "$CMD" = "DISCONNECTED" ] ; then

echo "event $CMD ignored because Connman is started"

exit 0

fi

fi

if [ "$CMD" = "CONNECTED" ]; then

kill\_daemon udhcpc /var/run/udhcpc-$IFNAME.pid

#udhcpc -i $IFNAME -p /var/run/udhcpc-$IFNAME.pid -S

ifconfig $IFNAME 10.20.0.13 netmask 255.0.0.0

fi

if [ "$CMD" = "DISCONNECTED" ]; then

kill\_daemon udhcpc /var/run/udhcpc-$IFNAME.pid

ifconfig $IFNAME 10.20.0.13 netmask 255.0.0.0

fi

if [ "$CMD" = "P2P-GROUP-STARTED" ]; then

GIFNAME=$3

if [ "$4" = "GO" ]; then

kill\_daemon udhcpc /var/run/udhcpc-$GIFNAME.pid

ifconfig $GIFNAME 192.168.42.1 up

cp /etc/wpa\_supplicant/udhcpd-p2p.conf /etc/wpa\_supplicant/udhcpd-p2p-itf.conf

sed -i "s/INTERFACE/$GIFNAME/" /etc/wpa\_supplicant/udhcpd-p2p-itf.conf

udhcpd /etc/wpa\_supplicant/udhcpd-p2p-itf.conf

fi

if [ "$4" = "client" ]; then

kill\_daemon udhcpc /var/run/udhcpc-$GIFNAME.pid

kill\_daemon udhcpd /var/run/udhcpd-$GIFNAME.pid

udhcpc -i $GIFNAME -p /var/run/udhcpc-$GIFNAME.pid

fi

fi

if [ "$CMD" = "P2P-GROUP-REMOVED" ]; then

GIFNAME=$3

if [ "$4" = "GO" ]; then

kill\_daemon udhcpd /var/run/udhcpd-$GIFNAME.pid

ifconfig $GIFNAME 255.0.0.0

fi

if [ "$4" = "client" ]; then

kill\_daemon udhcpc /var/run/udhcpc-$GIFNAME.pid

ifconfig $GIFNAME 255.0.0.0

fi

fi

root@edison04:/etc/wpa\_supplicant# wpa\_cli save

//Ctrl-o Enter Ctrl-x

root@edison:/lib/systemd/system# iwconfig wlan0 power off

root@edison:/lib/systemd/system# systemctl start wifi-power-management-off

//setup network

root@edison:/lib/systemd/system# wpa\_cli -iwlan0 remove\_network all

OK

root@edison:/lib/systemd/system# wpa\_cli save

Selected interface 'wlan0'

OK

root@edison:/lib/systemd/system# systemctl start connman

root@edison:/lib/systemd/system# systemctl enable connman

ln -s '/lib/systemd/system/connman.service' '/etc/systemd/system/multi-user.target.wants/connman.service'

root@edison:/lib/systemd/system# connmanctl

connmanctl> scan wifi

Scan completed for wifi

connmanctl> services

TP-LINK\_3BED wifi\_00aefa09a3c4\_54502d4c494e4b5f33424544\_managed\_psk

//setup static ip

//usage: connmanctl>config <service> --ipv4 off

connmanctl>config wifi\_00aefa09a3c4\_54502d4c494e4b5f33424544\_managed\_psk --ipv4 off

connmanctl> agent on

Agent registered

//usage connmanctl>config *<service>* --ipv4 manual *<ip\_address> <netmask> //<gateway>*

connmanctl>config wifi\_00aefa09a3c4\_54502d4c494e4b5f33424544\_managed\_psk--ipv4 manual *10.20.0.x 255.0.0.0 10.20.0.1*

connmanctl> connect wifi\_00aefa09a3c4\_54502d4c494e4b5f33424544\_managed\_psk

*//input passphrase for wifi*

*//crtl+c*

root@edison:/lib/systemd/system# wpa\_cli -iwlan0 save