

Intel Do-It-Yourself Challenge Rebuild (with) Yocto

Nicolas Vailliet

www.Intel-Software-Academic-Program.com

paul.guermonprez@intel.com

Intel Software

2014-02-01



Why ? Should I ?

Yocto ?

Is Yocto a distribution ?

*"Yocto is not an embedded Linux distribution
it creates a custom one for you"*

<https://www.yoctoproject.org/about>

What's included ?

*"Yocto provides templates, tools and methods to help you
create custom Linux-based systems for embedded products
regardless of the hardware architecture."*



Rebuild (with) Yocto

Prepackaged images

Yocto for Galileo comes with several prepackaged images you can download and boot from using a microCD card. And you are free to install packages manually after booting the image on the Galileo.

Rebuild the OS

But what if you want to rebuild the full OS yourself? Perhaps you want to change what's installed by default, or change kernel parameters (may be required for some IO operations).

Even if you don't need it, it's always nice to know you can. That's Linux !



Yocto output files

Rebuild Yocto vs build for Yocto

Yocto can be used in two different ways. After a short configuration phase, you will choose between generating a Linux file system file or the cross compile development environment (or toolchain).



And then, it will take up to 8 hours to compile output files, on a notebook (depending on system configuration and network connection speed).

On a Sandy bridge workstation, it lasts less than 2 hours.

You'll need

Board Support Package Sources for Intel Quark

An archive with all tools you need, available on Galileo drivers website.

<https://communities.intel.com/docs/DOC-22226>

Storage and internet bandwidth

Make sure you have 100Gb available on your hard drive. We'll also need a good internet connectivity.

CPU

If you would like to rebuild the full Yocto OS or generate the cross compile toolchain, it will take up to 8 hours on a workstation.



You'll need

Linux

We'll use Ubuntu 12.04

Packages

```
"sudo apt-get install build-essential sed wget cvs  
subversion git-core coreutils unzip texi2html texinfo  
libsdl1.2-dev docbook-utils gawk python-pysqlite2 diffstat  
help2man make gcc g++ desktop-file-utils chrpath libgl1-  
mesa-dev libglu1-mesa-dev mercurial autoconf automake  
groff libtool xterm p7zip-full bitbake"
```



Rebuilding (with) Yocto

Procedure (1/5)

Uncompress

```
7z x Board_Support_Package_Sources*.7z
```

Rename your BSP folder with a shorter name

```
mv Board_Support_Package_*** BSP_Galileo
```

Unpack

```
tar xvzf meta-clanton_*.tar.gz
```

Go to the new Yocto home folder

```
cd meta-clanton_...
```

Download, compile and set up Poky, a tool used by Yocto

```
./setup.sh  
source poky/oe-init-build-env yocto_build
```



Procedure (2/5)

Setting up Yocto recipe

By default, Yocto is ready to compile a tiny Linux image or cross compile toolchain. But, we want to compile, debug and use libraries.

Asking for a full configuration:

Edit the conf/local.conf file.

Change "clanton-tiny" to "clanton-full".

Set BB_NUMBER_THREADS and PARALLEL_THREADS to "number of cores your processor has multiply by 3".

Save the file.



Procedure (3/5)

Disable uClibc

It will disable uClibc, and replace it by EGlibc, which have more features and is commonly used under Linux.

Edit `"../meta-clanton-distro/recipes-multimedia/v4l2apps/v4l-utils_0.8.8.bbappend"`

Comment these 3 lines:

```
#FILESEXTRAPATHS_prepend := "${THISDIR}/files:"  
#SRC_URI += file://uclibc-enable.patch  
#DEPENDS += "libiconv"
```



Procedure (4/5)

Get a default config

Copy the full image configuration from a sample:

```
cp ../meta-clanton-distro/recipes-core/images/image-full.bb ../meta-clanton-distro/recipes-core/images/image-sdk.bb
```

Edit the default config

These options are set up for generating the associated Linux image later.

You can (un)comment features you do (not) want. The image size will be 3GB.

Edit ../meta-clanton-distro/recipes-core/images/image-sdk.bb :

```
IMAGE_INSTALL = "packagegroup-core-boot ${ROOTFS_PKGMANAGE_BOOTSTRAP}  
${CORE_IMAGE_EXTRA_INSTALL} packagegroup-core-basic packagegroup-core-lsb  
kernel-dev"
```

```
IMAGE_FEATURES += "package-management tools-sdk dev-pkgs tools-debug  
eclipse-debug tools-profile tools-testapps debug-tweaks"
```

```
IMAGE_ROOTFS_SIZE = "3072000"
```



Procedure (5/5)

Last details

To have a Linux system that can support full Galileo connectivity (pins, pwm...), you have to apply the following patch:

Edit `../meta-clanton-bsp/recipes-kernel/linux/files/clanton.patch`

```
+static unsigned int i2c_std_mode = 1;
```

This is line #10722. By default, this variable is not initialized. You set it to 1.

Configure kernel features

```
bitbake linux-yocto-clanton -c menuconfig
```

Launch the incredible machine

It will compile the cross compile toolchain.

This is the step that will take a while....

```
bitbake image-sdk
```



How to use Linux image

How to use Linux image files

Get Yocto output files

Files you generated are in: `yocto_build/tmp/deploy/images`.

Deploy on your microSD card

Copy the following files and folder on a SD card:

- `bzImage-*` (rename it to `bzImage`)
- `core-image-minimal-initramfs-*` (rename it to `core-image-minimal-initramfs-clanton.cpio.gz`)
- `image-*` (rename it to `image-full-clanton.ext3`)
- boot folder

Insert the SD card in Galileo board slot

Then, boot it! Ethernet configuration is automatic.

Keep your Board Support folder if you plan to generate other output files with Yocto.



Tips & Troubleshooting

- On malitov.com, Sergey's blog explains how to add multiple image on the SD card (that could be use useful).
- If you run bitbake via remote access, you will have the following error:

ERROR: No valid terminal found, unable to open devshell

Try to connect to your machine with `ssh -X` to enable X forwarding.

If it doesn't work, don't use `'-c menuconfig'` attribute. (you won't be able to customize the kernel using the config menu).

- If `do_compile` Node JS can fail with the following error:

Make[1]: Execvp: printf: Argument list too long

To solve this problem, be sure `'pwd'` command prints less than 50 characters in your shell when you are in yocto-build folder.



Tips & Troubleshooting

- You can edit and customize the Linux file system before booting on the board.

To do so, insert the SD card in your computer and run the following command:

```
Sudo mount -o loop -t ext3 /media/yoursdcard/image-full-clanton.ext3 /mnt
```

You can now access the card sysfs as if you were running it.

It's very useful to edit the Ethernet configuration to use a static IP or to copy a new Wifi driver...

See next slide to set up Wifi with this full image.



Tips & Troubleshooting

- The full Linux image contains the connection manager, which is tool to manage wifi, ethernet and bluetooth connection.
- It allows only one established connection by default. When it's available Wifi is default. Otherwise, Ethernet is used.
- However, you can use both with the command `>ifup eth0`.
- We recommend to configure your router to associate a static IP address for your board (one for each technology).
- Follow next slide procedure to install the previously described behaviour.



Tips & Troubleshooting

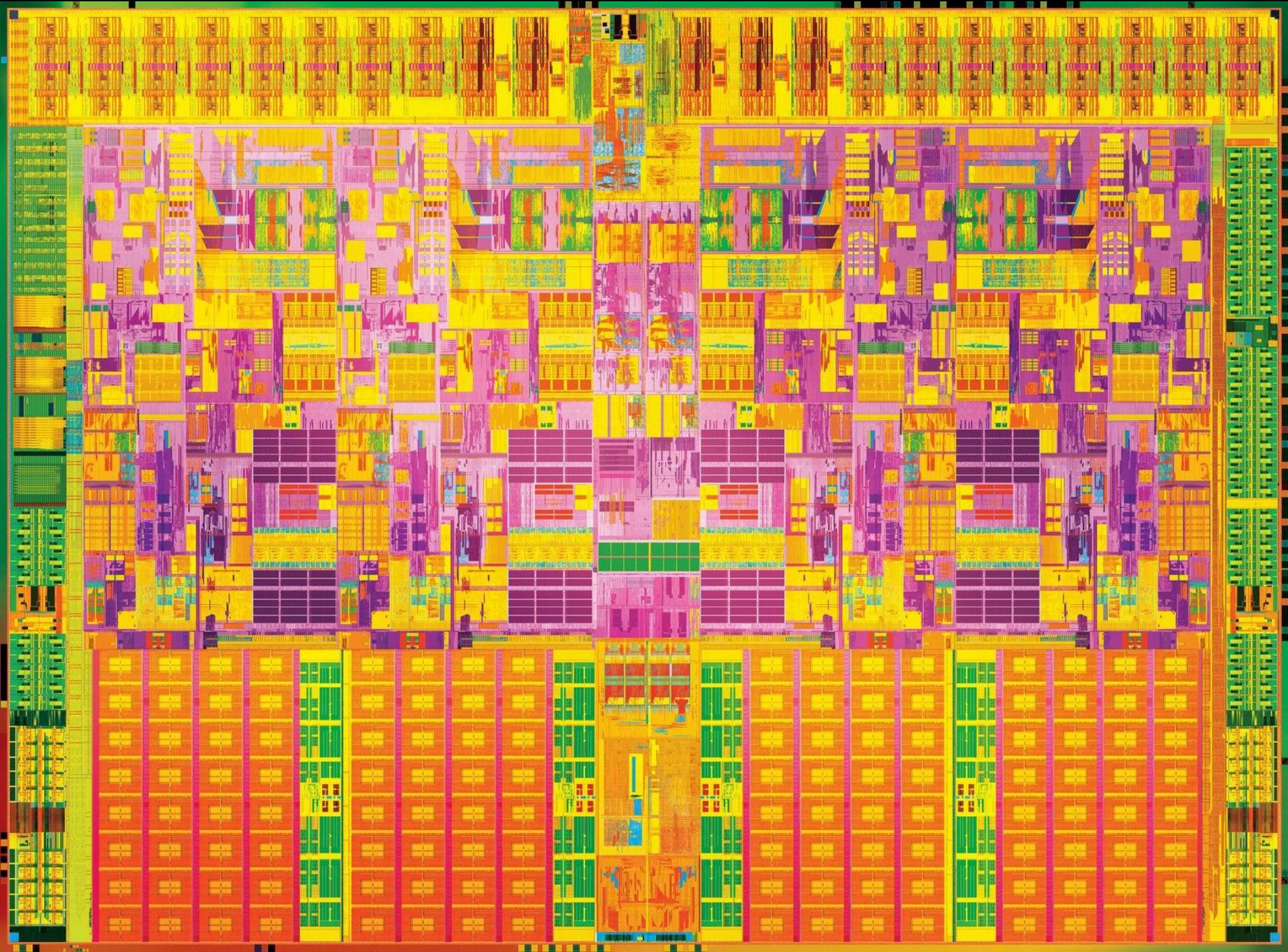
- To set up Wifi access to a WPA-secured hotspot:
- `>connmanctl`
- `>enable wifi`
- `>scan wifi`
- `>services` (copy/paste your access point ID `wifi_***_psk`)
- `>exit`
- `>cat << EOF > /var/lib/connman/wifi.config`
- `[service_wifi_*****_managed_psk]`
- `Type = wifi`

Tips & Troubleshooting

- >Security = wpa
- >Name = ESSID
- >Passphrase = WPA_KEY
- >EOF
- Reboot the board
- other ways to do it are:
- >/etc/init.d/connman restart
- OR
- >connmanctl
- >connect wifi_*****_managed_psk
- OR ifup wlan0

External links

- <http://ph0b.com/adding-dev-tools-gcc-make-to-galileo-sd-image/>
- <http://www.malinov.com/Home/sergey-s-blog>
- <http://www.cnx-software.com/2012/02/09/yocto-project-quick-start-guide-for-ubuntu/>



License Creative Commons – By 3.0

You are free:

- **to Share** — to copy, distribute and transmit the work
- **to Remix** — to adapt the work
- to make commercial use of the work

Under the following conditions:

- **Attribution** — You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

With the understanding that:

- **Waiver** — Any of the above conditions can be waived if you get permission from the copyright holder.
- **Public Domain** — Where the work or any of its elements is in the public domain under applicable law, that status is in no way affected by the license.
- **Other Rights** — In no way are any of the following rights affected by the license:
 - Your fair dealing or fair use rights, or other applicable copyright exceptions and limitations;
 - The author's moral rights;
 - Rights other persons may have either in the work itself or in how the work is used, such as publicity or privacy rights.
- **Notice** — For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this web page.

<http://creativecommons.org/licenses/by/3.0/>