Booting It Successfully
For The First Time In Mainline
Enric Balletbò i Serra

Electronics Engineer

- Hardware and Linux enthusiast
- Kernel contributor
- More than 10 years of experience bringing-up different kind of boards
Enric Balletbò i Serra

Electronic Engineer

Hardware and Linux enthusiast
Kernel contributor
More than 10 years of experience bringing-up different kinds of boards.
Agenda

- **What is Board Bring-Up?**
- Bring-Up process of a new board
- What’s the problem
- How software can help us with ...
- Conclusions
- Lessons learned
What is Board Bring-Up?

“Board bring-up is the process of validating, both electrically and functionally, a new circuit board design including the porting of boot firmware and the development of a Board Support Package.”
Agenda

- What is Board Bring-Up?
- Bring-Up process of a new board
- What’s the problem
- How software can help us with ...
- Conclusions
- Lessons learned
Bring-Up process of a new board

- Component selection
- Circuit Design (Schematic)
- Bill of Materials (BOM)
- PCB prototypes
- Assembly verification
- Basic hardware test
Bring-Up process of a new board

- Component selection
- **Circuit Design (Schematic)**
- Bill of Materials (BOM)
- PCB prototypes
- Assembly verification
- Basic hardware test
Bring-Up process of a new board

- Component selection
- Circuit Design (Schematic)
- **Bill of Materials (BOM)**
- PCB prototypes
- Assembly verification
- Basic hardware test
Bring-Up process of a new board

- Component selection
- Circuit Design (Schematic)
- Bill of Materials (BOM)
- **PCB prototypes**
- Assembly verification
- Basic hardware test

**Hardware**
Bring-Up process of a new board

- Component selection
- Circuit Design (Schematic)
- Bill of Materials (BOM)
- PCB prototypes
- **Assembly verification**
- Basic hardware test
Bring-Up process of a new board

- Component selection
- Circuit Design (Schematic)
- Bill of Materials (BOM)
- PCB prototypes
- Assembly verification
- Basic hardware test

Hardware
Bring-Up process of a new board

- Firmware programming
- Boot-loader
- Kernel and device drivers
- Software stack
- Application development
Bring-Up process of a new board

- Firmware programming
- **Boot-loader**
- Kernel and device drivers
- Software stack
- Application development

Software
Bring-Up process of a new board

- Firmware programming
- Boot-loader
- **Kernel and device drivers**
- Software stack
- Application development

Software
Bring-Up process of a new board

- Firmware programming
- Boot-loader
- Kernel and device drivers
- **Software stack**
- Application development
Bring-Up process of a new board

- Firmware programming
- Boot-loader
- Kernel and device drivers
- Software stack
- Application development

Software
Agenda

- What is Board Bring-Up?
- Bring-Up process of a new board
- What’s the problem
- How software can help us with ...
- Conclusions
- Lessons learned
What's the problem?
What's the problem?
What’s the problem?
What’s the problem?
What's the problem?

"No, you back off! I was here before you!"
And how we can solve this?

Software

Hardware
And how we can solve this?

Software

Hardware

Work together from the beginning!
Agenda

- What is Board Bring-Up?
- Bring-Up process of a new board
- What’s the problem
- How software can help us with ...
- Conclusions
- Lessons learned
How software can help us with: Component Selection

- Main processor
  - Evaluate the Board Support Package
  - Evaluate mainline support
- Other IC (sensors, displays controller, real time clocks, etc)
  - Check that the driver is upstream
- Extra note
  - Do NOT overuse microcontrollers
# Vendor vs Mainline

## Texas Instruments

![Graph showing number of commits and files changed for different versions of the texas instruments.](image)

<table>
<thead>
<tr>
<th>Vendor branch</th>
<th>Stable version</th>
<th>Number of commits</th>
<th>Files changed</th>
<th>Insertions</th>
<th>Deletions</th>
</tr>
</thead>
<tbody>
<tr>
<td>ti-isk-linux-4.4</td>
<td>4.4.113</td>
<td>3084</td>
<td>1526</td>
<td>192339</td>
<td>28440</td>
</tr>
<tr>
<td>ti-isk-linux-4.9</td>
<td>4.9.83</td>
<td>2103</td>
<td>995</td>
<td>119505</td>
<td>16509</td>
</tr>
</tbody>
</table>
How software can help us with:

Circuit Design (Schematic)

- Let software team review the schematic
  - Write the device tree file
  - Special attention with:
    - Muxer settings (power management)
    - Enable of ICs and regulators
    - Lines crossed (TX/RX, MISO/MOSI, etc)
    - I2C addresses
Connecting two displays (First option)

Hardware
Connecting two displays (First option)

Aplication

kernel driver

μC (Firmware)

fb0

fb1

Display

Display

Software
Connecting two displays (Second option)

Hardware
Connecting two displays (Second option)

Software
How software can help us with: *Bill Of Materials (BOM)*

- An alternative part might have different software support
How software can help us with: *PCB Prototypes*

- Time to plan your tests?
  - Stress testing
  - Performance testing
  - Connectivity testing
  - Functional testing
- Manual testing or automated testing
How software can help us with: 
**Assembly Verification**

- Run stress and throughput tests
  - RAM memory
  - Non-volatile memory
  - Buses (USB, SPI, I2C, etc)
- Define a test case for all that is wired
How software can help us with: **Basic Hardware Test**

- Functional testing
  - Against the business requirements of application
- User Acceptance testing
  - Verify that a solution works for the user
- Regression testing
  - Every time you do a hardware modification
Agenda

• What is Board Bring-Up?
• Bring-Up process of a new board
• What’s the problem
• How software can help us with ...
• Conclusions
• Lessons learned
Booting it successfully for the first time with mainline

- **NO SECRET**
  - Your hardware must be supported in mainline
  - Component selection matters

- **Upstream first**
  - Upstream it before you get the first prototypes
Agenda

• What is Board Bring-Up?
• Bring-Up process of a new board
• What’s the problem
• How looking at software can help us with ...
• Conclusions
• Lessons learned
Lessons learned

- Review the schematic from software point of view can catch lots of errors

- Try to develop the software as much as possible even before you have the first prototypes

- Before doing a respin, test for regressions
Thank you!