



Devicetree Hardware Autoconfiguration

Presented by
Hans de Goede

Introducing myself

- Software Engineer working for Red Hat on USB, human input devices and Xorg
- Working on u-boot and kernel support for Allwinner SoCs in my spare time

Today's Topics

1. Hardware autoconfig on q8 tablets
2. Hardware autoconfig in general
3. Conclusion
4. Q & A and discussion

Q8 Tablets

Q8 tablets

- q8 (aka q88) is a generic 7" tablet enclosure
- 40 usd for 1024x600 lcd quadcore 512MB RAM
- SoC Allwinner A13 / A23 / A33 (mostly)
- The LCD + SoC / RAM / NAND are standard
- Every other batch uses a different touchscreen, accelerometer and wifi-chip
- Question of the day: is it possible to support all of these with a single dtb / u-boot (per SoC) ?

Touchscreen

Touchscreen detection



- 4 models touchscreen:
 - Silead gsl1680 revision 1
 - Silead gsl1680 revision 2
 - Elan eKTF2127
 - Zeitec zet6251
- Detected by prodding the i2c bus
- gsl1680 has an id register which allows telling apart the 2 revisions

Touchscreen problems



- The Silead gsl1680 is a flexible little ship:
 - Configurable pin muxing allowing flexible digitizer pin routing
 - Different PCB's with this chip need different firmware files
 - x/y coordinates may be inverted vs screen orientation
 - Reported resolution is firmware configurable
- Problem: The needed firmware and x/y inversion cannot be detected

Touchscreen solution

- Per gsl revision we can cover all known tablets with 2 firmware files
- So far the touchscreen + accelerometer combination seems to be unique:
 - Pick firmware-name based on detected accelerometer
 - Set resolution + swap x/y to match
 - Module parameters to allow overriding all of these
 - dev_warn that they user may need to override things
 - dev_info current settings

Accelerometer

Accelerometer

- Many models: da226, da280, da311, dmard06, dmard09, dmard10, mc3230, mma7660, mxc6225
- Detected by prodding the i2c bus
- Problem: orientation
- Most accelerometers seem to be used with a fixed orientation
- For those which aren't, pick a default orientation based on SoC + touchscreen
- Module parameter to allow overriding orientation

End Result

End Result

```
[screen 0: ttyACM0]
[root@localhost ~]# dmesg | grep q8
[ 1.488427] q8-hardwaremgr q8-hardwaremgr.0: Looking for touchscreen without
a regulator
[ 1.526197] q8-hardwaremgr q8-hardwaremgr.0: Found Silead touchscreen ID: 0xb
4820000
[ 1.534225] q8-hardwaremgr q8-hardwaremgr.0: Looking for accelerometer withou
t a regulator
[ 1.545036] q8-hardwaremgr q8-hardwaremgr.0: Found miramems.da312 acceleromet
er at 0x26
[ 1.553087] q8-hardwaremgr q8-hardwaremgr.0: gsl1680 touchscreen may require
kernel cmdline parameters to function properly
[ 1.564246] q8-hardwaremgr q8-hardwaremgr.0: Try q8_hardwaremgr.touchscreen_i
nvert_x=1 if x coordinates are inverted
[ 1.574792] q8-hardwaremgr q8-hardwaremgr.0: Try q8_hardwaremgr.touchscreen_v
ariant=1 if coordinates are all over the place
[ 1.585950] q8-hardwaremgr q8-hardwaremgr.0: touchscreen_variant 0 (auto)
[ 1.592758] q8-hardwaremgr q8-hardwaremgr.0: touchscreen_width 960 (auto)
[ 1.599568] q8-hardwaremgr q8-hardwaremgr.0: touchscreen_height 640 (auto)
[ 1.606465] q8-hardwaremgr q8-hardwaremgr.0: touchscreen_invert_x 0 (auto)
[ 1.613359] q8-hardwaremgr q8-hardwaremgr.0: touchscreen_invert_y 1 (auto)
[ 1.620253] q8-hardwaremgr q8-hardwaremgr.0: touchscreen_swap_x_y 0 (auto)
[ 1.627150] q8-hardwaremgr q8-hardwaremgr.0: touchscreen_fw_name gsl1680-b482
-q8-d702.fw (auto)
[root@localhost ~]#
```

General HW Autoconfiguration

General HW autoconfig



- If you're in control of the hardware:
- Make sure you can uniquely identify the HW
- Add a way to get the board revision from the HW
- If there are add-ons add a way to enumerate add-ons
- Add-ons should be versioned too
- If you're not in control of the hardware:
- Good luck, you're going to need it

Applying DT changes



- If you've a fixed set of modifications to apply, consider using dt overlays and an in kernel overlay-manager
- If you've lots of small separate settings which together form a huge number of possible combinations, consider using devicetree changesets, with an in-kernel hw-manager

DT changeset example



```
struct of_changeset cset;
struct device_node *np;

np = of_find_node_by_name(of_root, "touchscreen");
of_changeset_init(&cset);
of_changeset_add_property_u32(&cset, np, "reg", 0x40);
of_changeset_add_property_string(&cset, np, "compatible",
                                "silead,gsl1680");
of_changeset_update_property_string(&cset, np, "status",
                                    "okay");
of_changeset_apply(&cset);
of_node_put(np);
```


Conclusion

Conclusion

- Yes it is possible to use hw-autoconfig for these cheap tablets
- Do-s and don't-s learned:
 - Do start thinking about this during your hardware design
 - Do not leave this till after the hardware is complete
 - Do make sure your hw revisions / variants are easily uniquely identifiable
 - Don't add a node to devicetree representing your manager; instead bind your manager to the board compatible

Questions?



Contact:

hdegoede@redhat.com

Git repositories:

<https://github.com/jwrdegoede/>

This work is licensed under a [Creative Commons Attribution-ShareAlike 3.0 Unported License](https://creativecommons.org/licenses/by-sa/3.0/)