C.H.I.P. The world's first nine dollar computer

Presented by
Hans de Goede

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 Unported License
Today's Topics

1. Introducing myself
2. Introducing the C.H.I.P.
3. C.H.I.P. accessories
4. C.H.I.P. upstream support status
5. Demo?
6. Questions
Introducing myself
Introducing myself

- Software Engineer working for Red Hat on USB, human input devices and nouveau
- Not affiliated with Next Thing Co., the makers of the C.H.I.P. in any way
- Working on u-boot and kernel support for Allwinner SoCs in my spare time
Introducing the C.H.I.P.
The C.H.I.P.

- A full computer for $9 in 60mm x 41mm
R8 Module

- Allwinner R8 SoC
  - 1GHz Cortex A8
  - Mali400 GPU
- 512MB DDR3 RAM
- 4 GB Nand flash storage
RTL8723BS wifi/bt

- 802.11b/g/n 1/1 2.4 GHz
- Bluetooth 4.0
Connectors

- USB-A connected to an EHCI/OHCI controller pair
- 3.5mm headphone jack:
  - Stereo audio out
  - Microphone in or composite video out selected by jumper
- Micro USB-B USB-2 OTG
- LiOn / LiPo battery connector
### ALPHA C.H.I.P. (v0.21) PINOUT

<table>
<thead>
<tr>
<th>U13</th>
<th>U14</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GND</strong></td>
<td><strong>GND</strong></td>
</tr>
<tr>
<td>VCC-5V</td>
<td>UART1-TX</td>
</tr>
<tr>
<td>VCC3V3</td>
<td>UART1-RX</td>
</tr>
<tr>
<td>VCC-1V8</td>
<td>UBOOT</td>
</tr>
<tr>
<td>TWI1-SDA</td>
<td>LRADC</td>
</tr>
<tr>
<td>TWI1-SCK</td>
<td><strong>GND</strong></td>
</tr>
<tr>
<td>X1</td>
<td>X2</td>
</tr>
<tr>
<td>Y1</td>
<td>Y2</td>
</tr>
<tr>
<td>LCD-D2</td>
<td>PWM0</td>
</tr>
<tr>
<td>LCD-D4</td>
<td>LCD-D3</td>
</tr>
<tr>
<td>LCD-D6</td>
<td>LCD-D5</td>
</tr>
<tr>
<td>LCD-D10</td>
<td>LCD-D7</td>
</tr>
<tr>
<td>LCD-D12</td>
<td>LCD-D11</td>
</tr>
<tr>
<td>LCD-D14</td>
<td>LCD-D13</td>
</tr>
<tr>
<td>LCD-D18</td>
<td>LCD-D15</td>
</tr>
<tr>
<td>LCD-D20</td>
<td>LCD-D19</td>
</tr>
<tr>
<td>LCD-D22</td>
<td>LCD-D21</td>
</tr>
<tr>
<td>LCD-CLK</td>
<td>LCD-D23</td>
</tr>
<tr>
<td>LCD-VSYNC</td>
<td>LCD-HSYNC</td>
</tr>
<tr>
<td><strong>GND</strong></td>
<td><strong>GND</strong></td>
</tr>
</tbody>
</table>

### Ports
- **U13**
  - CHG-IN
  - GND
  - TS
  - BAT
  - PWRON
  - GND
  - X2
  - Y2
  - LCD-D3
  - LCD-D5
  - LCD-D7
  - LCD-D11
  - LCD-D13
  - LCD-D15
  - LCD-D19
  - LCD-D21
  - LCD-D23
  - LCD-HSYNC
  - LCD-DE
- **U14**
  - VCC-5V
  - HPL
  - HPCOM
  - HPR
  - MICM
  - MICIN1
  - XIO-P0
  - XIO-P1
  - XIO-P3
  - XIO-P5
  - XIO-P7
  - GND
  - AP-EINT1
  - AP-EINT3
  - TWI2-SDA
  - TWI2-SCK
  - CSIPCK
  - CSICK
  - CSIVSYNC
  - CSID0
  - CSID1
  - CSID2
  - CSID3
  - CSID4
  - CSID5
  - CSID6
  - CSID7
  - GND
Alternative header use

- LCD pins D2-D5: UART2
- LCD pins other: 100Mbit eth (with external phy)
- CSI clks + sync: SPI2
- CSI D0-D5: MMC2
C.H.I.P. Accessories
C.H.I.P. Accessories
PocketC.H.I.P.
Use a Pencil as a Kickstand

Attach a Shoelace and Carry PocketChip Anywhere

Tough Injection Molded Case

Built In GPIO Breakouts
C.H.I.P. Upstream support status
U-boot support

- Upstream u-boot fully supports the C.H.I.P.
- Except for the NAND flash
  - The first-stage loader (SPL) supports loading the second stage (u-boot) from NAND already
  - But u-boot is missing a full MTD nand driver for UBI(FS) access to load the kernel, dtb, etc.
- Next Thing Co.'s github u-boot repository does have preliminary support for this
Linux support

- Upstream Linux supports most of the C.H.I.P.
- Not supported yet are:
  - NAND
  - Wifi/Bt module
  - Hardware video encoding / decoding engine
  - GPU
  - Video output
Linux support

- NAND: Boris Brezillon from Free Electrons is working on this. Next Thing Co.'s github linux repository has preliminary support.

- Wifi/Bt: There is an out of tree driver for this: https://github.com/hadess/rtl8723bs. Hopefully this can be added to drivers/staging soon.
Linux support

- Hardware video encoding / decoding: this has been reverse-engineered but no one is working on a driver. The plan is to have an out-of-tree kernel driver which allows using Allwinner's userspace binaries for this.

- GPU: The plan is to have an out-of-tree kernel driver which allows using ARM's userspace binaries for this.
Video output

- U-Boot has video output support and the kernel can take over the framebuffer through simplefb
- Maxime Ripard from Free Electrons is working on a kms driver
Demo ?
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