FreeBSD support for OMAP3 BeagleBoard-xM

Aleksander Dutkowski
aleek@FreeBSD.org

October 20, 2012

Agenda

- About me
- What is Google Summer of Code
- BeagleBoard and OMAP review
- GSoC project
  - issues
- Current status of FreeBSD support
- OMAP QEMU
- future development

About me

- Aleksander Dutkowski
- 23 years old
- 5th (last) year of studying at Gdansk University of Technology (Computer Science, Computer Networks)
- embedded systems, kernel design and implementation, computer networks theory
- Google Summer of Code 2012
- Vector sp. z o.o.

Google Summer of Code

- annual, sincle 2005
- 3 months of coding
- the aim – popularize opensource, link new people with projects
- 1 212 students from all around the world
- 180 orgs
- 15 FreeBSD projects this year (including 2 in FreeBSD/arm branch)
- about 89% of students finished their projects
- http://wiki.freebsd.org/SummerOfCode2012
- Rafal Jaworowski - mentor

How does it look like?

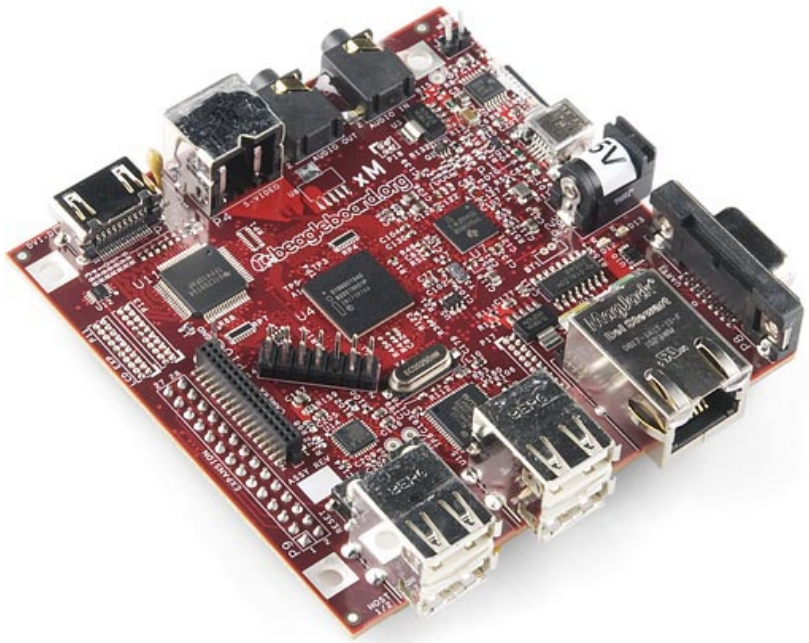

Figure: BeagleBoard-xM rev C

Former

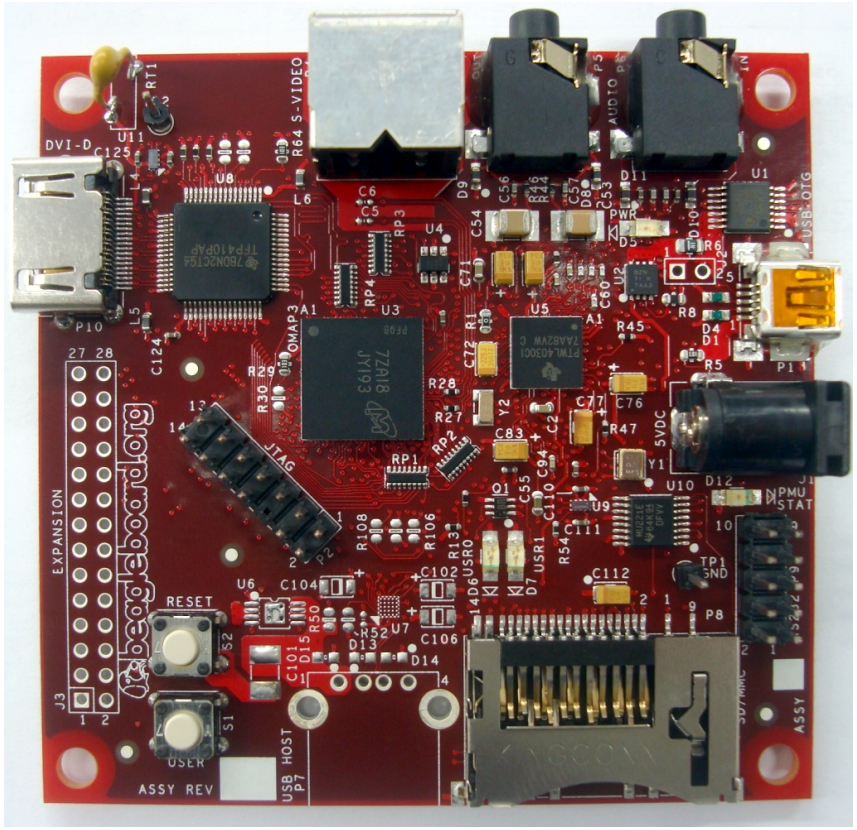

Figure: BeagleBoard - former of -xM version

Why BeagleBoard-xM?

- consultations with *cognet* and *dmarion*
- there was support for some ARM OMAP SoC's and OMAP based boards
- easier for beginner
- popularity and cheapness of the board

What was already done?

- some of omap generic code was already in *armv6* tree
- ports for PandaBoard and BeagleBone
- and of course Linux port done by BeagleBoard company

Internals

- TI DM3730 SoC (compatible with OMAP3530 (Cortex-A8), 800 MHz
- 512M RAM
- USB2.0 with onboard four-port HUB and 100 Ethernet
- DVI-D and LCD output
- SD card slot
- DSP included in SoC (C64x+)
- i2c, spi, i2s, mmc expansion
- audio in/out (TWL4030)
- BeagleBoard is software compatible with -xM version
- the same linux kernel for both

Writing code...

The project included writing/rewriting/adjusting following modules:

- mmc/sd driver
- gpio
- sDMA
- General Purpose Timers
- System Control Module
- Power Reset Control Module
- i2c
- usb ehci (smsc3320)
- TWL4030
- early uart support (side product)

Writing code...

The challenge wasn't to write the code – it was writing it *well*!

Writing code...

- inspecting the behavior of the chip
- reading linux kernel source
- reading similar code in other FreeBSD ports
- quality of TWL4030 driver in linux is very low :(

Flattened Device Tree

- used most in embedded world
- Typical problems, that FDT covers are:
  - memory layout
  - assignment of resources and identification on non-enumerable devices (i2c, spi buses, internal on-chip resources)
  - MAC-PHY binding
  - interrupts routing
  - GPIO pin routing and assignment

Flattened Device Tree

```
i2c: i2c@48070000 {
     compatible = "ti,i2c";
     reg =<  0x48070000 0x1000 >;
     interrupts = <56>;
     interrupt-parent = <&AINTC>;
     i2c-device-id = <1>;

     twl4030@48 {
          compatible = "ti,twl4030";
          reg = < 0x48 >;
          voltage-regulators =
               "vusb1v5", "1500",
               "vusb1v8", "1800",
               "vusb3v1", "3100",
               "vaux2", "1800";
     };
};
```

Issues

More interesting question is – what part of the project caused problems?

External non-linefetch abort

After enabling usb ehci stack, kernel started to panic

```
Fatal kernel mode data abort: 'External Non-Linefetch Abort (S)'
trapframe: 0xc05a6a14
FSR=00000008, FAR=c2e30848, spsr=400000d3
r0 =c057d2e0, r1 =00000004, r2 =00000004, r3 =c2e30848
r4 =c2e30800, r5 =00000100, r6 =c2e33cc0, r7 =00000302
r8 =c2e06c00, r9 =c025bb7c, r10=0000000c, r11=c05a6aa4
r12=c057d2e0, ssp=c05a6a60, slr=c02ddbbc, pc =c02ddc08

[ thread pid 0 tid 100000 ]
Stopped at      intr_event_add_handler+0xb8:    ldrb    r14, [r3, r15]
```

Figure: kernel panic after enabling omap ehci

External non-linefetch abort

```
(gdb) list *(intr_event_add_handler+0xb8)
0xc02ddc08 is in intr_event_add_handler (./machine/atomic.h:169).
164     static __inline u_int32_t
165     atomic_cmpset_32(...)
166     {
167             uint32_t ret;
168
169             __asm __volatile("1: ldrex %0, [%1]\n"
170                              "cmp %0, %2\n"
171                              "movne %0, #0\n"
172                              "bne 2f\n"
173                              "strex %0, %3, [%1]\n"
```

Figure: debugging the kernel

External non-linefetch abort

- allocing DMA tag by `malloc()`
- allocating DMA memory by `bus_dmamem_alloc()` with BUS_DMA_COHERENT so the *Cacheable* attribute is cleared
- both allocated chunks are within the page boundaries
- the whole page has *Cacheable* attr cleared
- DMA tag has tags, which are managed by atomic operations, `ldrex`, `strex`
- using `ldrex` on the memory page without *cacheable* attribute causes Data Abort Exception
- this bug wasn't on *qemu*
- regards to *gonzo*

External non-linefetch abort

```
if (flags & BUS_DMA_COHERENT)
    pmap_change_attr((vm_offset_t)*vaddr, len,
            BUS_DMA_NOCACHE);
```

Figure: head/sys/arm/arm/busdma_machdep-v6.c

issue with reseting USB

- SMSC USB3320 - USB 2.0 ULPI transceiver
- setting ULPI device includes setting GPIO line 0,then 1, and wait for some register to become cleared
- if the register is not cleared, the chip is hanged
- it seems that either usb setup is buggy, or u-boot is starting a transaction without finishing it
- possible solution is powering the device down and up. It is done via TWL4030 chip, which is communicating with OMAP via i2c
- I cant talk with TWL via i2c, because interrupts are disabled during bootup :(
- usb works with couple of peripherals
- SMSC LAN9512 causes Translation Fault?
- work on this bug is ongoing

qemu for omap

- there is a fork of qemu/arm which supports omap based boards
- supports BeagleBoard
- successfully ran freebsd kernel image

Advantages of this project

- adding support for OMAP3 based devices
- giving Beagle users choice
- one more kernel hacker ;)

Summary

- FreeBSD runs on BeagleBoard-xM
- finish and debug omap ehci and TWL4030 support
- add DVI chip support
- add LCD support
- DSP support?

Future

- ZedBoard support
  - Xilinx Zynq 7000 SoC with dual Cortex-A9 core
  - 512M DDR3
  - 1000 GbE
  - FPGA included!

Bibliography and sources

- http://code.google.com/soc
- http://wiki.freebsd.org/SummerOfCode2012/FreeBSDonBeagleBoardxM
- http://svnweb.freebsd.org/socsvn/soc2012/aleek/beaglexm-armv6/
- http://www.beagleboard.org/
- http://wiki.freebsd.org/FlattenedDeviceTree
- http://git.linaro.org/gitweb?p=qemu/qemu-linaro.git
- irc://irc.efnet.org/#bsdmips

Questions?

FreeBSD support for OMAP3 BeagleBoard-xM

Aleksander Dutkowski
aleek@FreeBSD.org

October 20, 2012