S3C24XX ARM Linux Overview

Introduction

The Samsung S3C24XX range of ARM9 System-on-Chip CPUs are supported by the 's3c2410' architecture of ARM Linux. Currently the S3C2410, S3C2412, S3C2413, S3C2416, S3C2440, S3C2442, S3C2443 and S3C2450 devices are supported.

Support for the S3C2400 and S3C24A0 series was never completed and the corresponding code has been removed after a while. If someone wishes to revive this effort, partial support can be retrieved from earlier Linux versions.

The S3C2416 and S3C2450 devices are very similar and S3C2450 support is included under the arch/arm/mach-s3c directory. Note, while core support for these SoCs is in, work on some of the extra peripherals and extra interrupts is still ongoing.

Configuration

A generic S3C2410 configuration is provided, and can be used as the default by *make s3c2410_defconfig*. This configuration has support for all the machines, and the commonly used features on them.

Certain machines may have their own default configurations as well, please check the machine specific documentation.

Layout

The core support files, register, kernel and paltform data are located in the platform code contained in arch/arm/mach-s3c with headers in arch/arm/mach-s3c/include

arch/arm/mach-s3c:

Files in here are either common to all the s3c24xx family, or are common to only some of them with names to indicate this status. The files that are not common to all are generally named with the initial cpu they support in the series to ensure a short name without any possibility of confusion with newer devices.

As an example, initially s3c244x would cover s3c2440 and s3c2442, but with the s3c2443 which does not share many of the same drivers in this directory, the name becomes invalid. We stick to s3c2440-<x> to indicate a driver that is s3c2440 and s3c2442 compatible.

This does mean that to find the status of any given SoC, a number of directories may need to be searched.

Machines

The currently supported machines are as follows:

Simtec Electronics EB2410ITX (BAST)

A general purpose development board, see EB2410ITX.txt for further details

Simtec Electronics IM2440D20 (Osiris)

CPU Module from Simtec Electronics, with a S3C2440A CPU, nand flash and a PCMCIA controller.

Samsung SMDK2410

Samsung's own development board, geared for PDA work.

Samsung/Aiji SMDK2412

The S3C2412 version of the SMDK2440.

Samsung/Aiji SMDK2413

The S3C2412 version of the SMDK2440.

Samsung/Meritech SMDK2440

The S3C2440 compatible version of the SMDK2440, which has the option of an S3C2440 or S3C2442 CPU module.

Thorcom VR1000

Custom embedded board

HP IPAQ 1940

Handheld (IPAQ), available in several varieties

HP iPAQ rx3715

S3C2440 based IPAQ, with a number of variations depending on features shipped.

Acer N30

A S3C2410 based PDA from Acer. There is a Wiki page at http://handhelds.org/moin/moin.cgi/AcerN30Documentation .

AML M5900

American Microsystems' M5900

Nex Vision Nexcoder Nex Vision Otom

Two machines by Nex Vision

Adding New Machines

The architecture has been designed to support as many machines as can be configured for it in one kernel build, and any future additions should keep this in mind before altering items outside of their own machine files.

Machine definitions should be kept in arch/arm/mach-s3c, and there are a number of examples that can be looked at.

Read the kernel patch submission policies as well as the Documentation/arm directory before submitting patches. The ARM kernel series is managed by Russell King, and has a patch system located at http://www.arm.linux.org.uk/developer/patches/ as well as mailing lists that can be found from the same site.

As a courtesy, please notify <ben-linux@fluff.org> of any new machines or other modifications.

Any large scale modifications, or new drivers should be discussed on the ARM kernel mailing list (linux-arm-kernel) before being attempted. See http://www.arm.linux.org.uk/mailinglists/ for the mailing list information.

I2C

The hardware I2C core in the CPU is supported in single master mode, and can be configured via platform data.

RTC

Support for the onboard RTC unit, including alarm function.

This has recently been upgraded to use the new RTC core, and the module has been renamed to rtc-s3c to fit in with the new rtc naming scheme.

Watchdog

The onchip watchdog is available via the standard watchdog interface.

NAND

The current kernels now have support for the s3c2410 NAND controller. If there are any problems the latest linux-mtd code can be found from http://www.linux-mtd.infradead.org/

For more information see Documentation/arm/samsung-s3c24xx/nand.rst

SD/MMC

The SD/MMC hardware pre S3C2443 is supported in the current kernel, the driver is drivers/mmc/host/s3cmci.c and supports 1 and 4 bit SD or MMC cards.

The SDIO behaviour of this driver has not been fully tested. There is no current support for hardware SDIO interrupts.

Serial

The s3c2410 serial driver provides support for the internal serial ports. These devices appear as /dev/ttySAC0 through 3.

To create device nodes for these, use the following commands

> mknod ttySAC0 c 204 64 mknod ttySAC1 c 204 65 mknod ttySAC2 c 204 66

GPIO

The core contains support for manipulating the GPIO, see the documentation in GPIO.txt in the same directory as this file.

Newer kernels carry GPIOLIB, and support is being moved towards this with some of the older support in line to be removed.

As of v2.6.34, the move towards using gpiolib support is almost complete, and very little of the old calls are left.

See Documentation/arm/samsung-s3c24xx/gpio.rst for the S3C24XX specific support and Documentation/arm/samsung/gpio.rst for the core Samsung implementation.

Clock Management

The core provides the interface defined in the header file include/asm-arm/hardware/clock.h, to allow control over the various clock units

Suspend to RAM

For boards that provide support for suspend to RAM, the system can be placed into low power suspend.

See Suspend.txt for more information.

SPI

SPI drivers are available for both the in-built hardware (although there is no DMA support yet) and a generic GPIO based solution.

LEDs

There is support for GPIO based LEDs via a platform driver in the LED subsystem.

Platform Data

Whenever a device has platform specific data that is specified on a per-machine basis, care should be taken to ensure the following:

1. that default data is not left in the device to confuse the driver if a machine does not set it at startup
2. the data should (if possible) be marked as __initdata, to ensure that the data is thrown away if the machine is not the one currently in use.

   The best way of doing this is to make a function that kmalloc()s an area of memory, and copies the __initdata and then sets the relevant device's platform data. Making the function *__init* takes care of ensuring it is discarded with the rest of the initialisation code:

   ```
    static __init void s3c24xx_xxx_set_platdata(struct xxx_data *pd)
    {
        struct s3c2410_xxx_mach_info *npd;

      npd = kmalloc(sizeof(struct s3c2410_xxx_mach_info), GFP_KERNEL);
      if (npd) {
         memcpy(npd, pd, sizeof(struct s3c2410_xxx_mach_info));
         s3c_device_xxx.dev.platform_data = npd;
      } else {
           printk(KERN_ERR "no memory for xxx platform data\n");
      }
   }
   ```

   ```
   Note, since the code is marked as __init, it should not be
   exported outside arch/arm/mach-s3c/, or exported to
   modules via EXPORT_SYMBOL() and related functions.
   ```

Port Contributors

Ben Dooks (BJD) Vincent Sanders Herbert Potzl Arnaud Patard (RTP) Roc Wu Klaus Fetscher Dimitry Andric Shannon Holland Guillaume Gourat (NexVision) Christer Weinigel (wingel) (Acer N30) Lucas Correia Villa Real (S3C2400 port)

Document Author

Ben Dooks, Copyright 2004-2006 Simtec Electronics