



ÁRBOL DE DISPOSITIVOS

Árbol De Dispositivos

Introducción

Árbol de dispositivos

Nodo de dispositivos

Formato de la imagen

Programa de carga

Casos de estudio

Árbol De Dispositivos

Introducción

- Objetivo
- Alcance
- Ficheros y herramientas

Árbol de dispositivos

Nodo de dispositivos

Formato de la imagen

Programa de carga

Casos de estudio

Introducción

Sistema



Arquitectura

- x86
 - iA32
 - iA32e

- ARM
- MIPS
- SPARC
- M68k
- Powerpc
- x86
- iA64
- etc

Fabricante

- Intel
- AMD

- Freescale
- Texas
- Motorola
- Intel
- SUN
- STM
- etc

Periféricos

- 8237 (DMA)
- 8253 (Timer)
- 8255 (PPI)
- 8259 (PIC)
- 8284 (RTC)
- 8288 (BUS)
- (integrados originales)

Dependiente de la misión del sistema

Inicialización

- BIOS (interfaz estándar)
- Cargador de SO

- UEFI
- U-Boot
- Barebox

Introducción

18/03/2011 Linux Kernel Mailing list

Linus Torvalds:

"Gaah. Guys, this whole ARM thing is a f*cking pain in the ass."

"...seems to be totally insane"

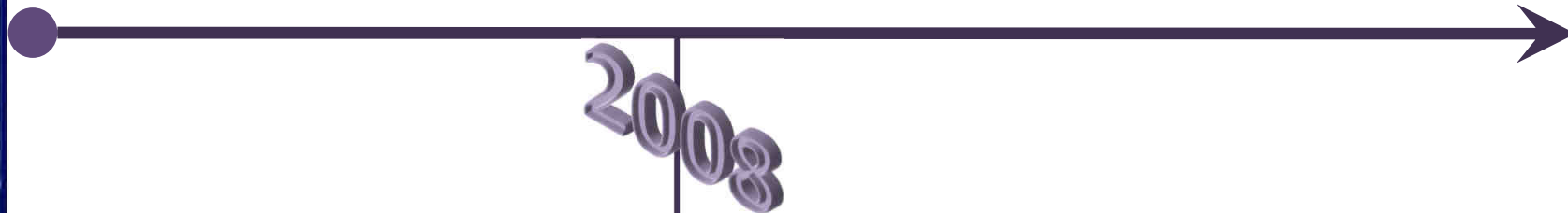
"...pure amount of crazy churn is annoying..."

"Somebody in the ARM community really needs to step up and tell people to stop dicking around."

"But the larger problem is that somebody really REALLY needs to think about how to get those crazy board details out of the kernel entirely."

Introducción

Nada nuevo bajo el Sol



Sun Microsystems - Open Boot / Open Firmware (1988) — Especificación ePAPR

- Used in SPARC systems

- Uses DT to describe hardware

IEEE-1275 formalized specification (1994)

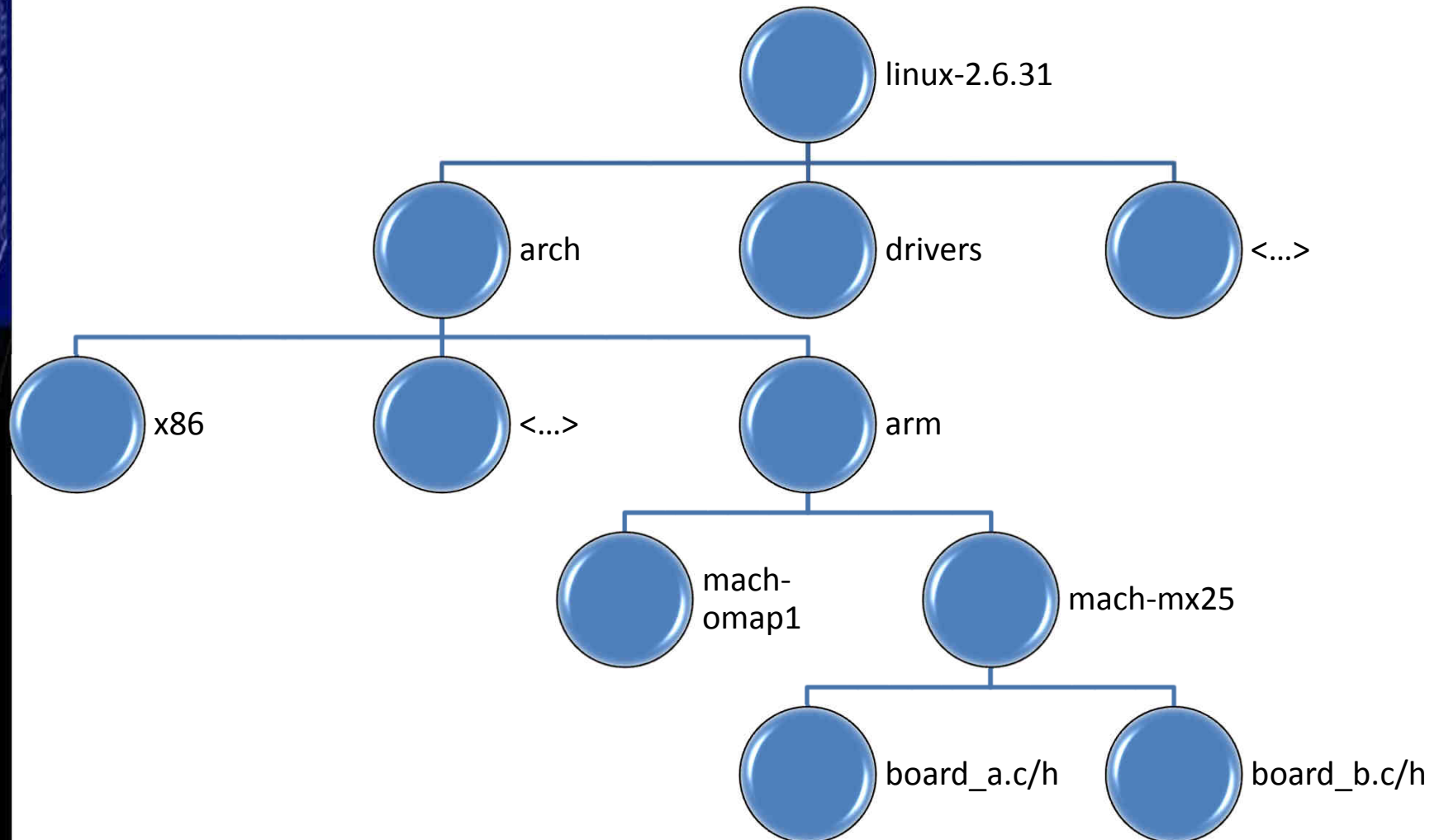
-

Documents DT

Apple adopts Open Firmware on Power Mac 7200 (1995) using DT

Common Hardware Reference Platform (CHRP) specifies DT (1995)

Introducción



Introducción

board-mx25-3stack.h

/ UART 5 configuration */*

#define UART5_MODE

MODE_DTE

#define UART5_IR

NO_IRDA

#define UART5_ENABLED

0

#define MXC_LL_UART_PADDR

UART1_BASE_ADDR

#define MXC_LL_UART_VADDR

AIPS1_IO_ADDRESS(UART1_BASE_ADDR)

/ Base address of debug board */*

#define DEBUG_BASE_ADDRESS

0x78000000

/ External ethernet LAN9217 base address */*

#define LAN9217_BASE_ADDR

DEBUG_BASE_ADDRESS

/ External UART */*

#define UARTA_BASE_ADDR

(DEBUG_BASE_ADDRESS + 0x08000)

#define UARTB_BASE_ADDR

(DEBUG_BASE_ADDRESS + 0x10000)

#define BOARD_IO_ADDR

0x20000

/ LED switches */*

#define LED_SWITCH_REG

(BOARD_IO_ADDR + 0x00)

/ buttons */*

#define SWITCH_BUTTON_REG

(BOARD_IO_ADDR + 0x08)

Introducción

mx25-3stack.c

```
/* working point(wp): 0 - 399MHz; 1 - 266MHz; 2 - 133MHz; */
```

```
static struct cpu_wp cpu_wp_mx25[] = {  
    {.pll_rate = 399000000,  
     .cpu_rate = 399000000,  
     .cpu_podf = 0x0,  
     .cpu_voltage = 1450000},  
    {.pll_rate = 532000000,  
     .cpu_rate = 266000000,  
     .cpu_podf = 0x1,  
     .cpu_voltage = 1340000},  
    {.pll_rate = 532000000,  
     .cpu_rate = 133000000,  
     .cpu_podf = 0x3,  
     .cpu_voltage = 1340000}, };
```

```
static struct keypad_data keypad_plat_data = {  
    .rowmax = 4,  
    .colmax = 4,  
    .learning = 0,  
    .delay = 2,  
    .matrix = keymapping,};
```

Introducción

Objetivo

- 1. ¿Cómo se le indica al SO sobre que hardware se ejecutará?*
- 2. ¿Cómo se puede modificar esa información sin que sea necesario reconstruir el kernel?*

Introducción

Objetivo

1. *¿Cómo se le indica al SO sobre que hardware se ejecutará?*
 - Utilizar una estructura estandarizada que describa el hardware.
2. *¿Cómo se puede modificar esa información sin que sea necesario reconstruir el kernel?*

Introducción

Objetivo

1. *¿Cómo se le indica al SO sobre que hardware se ejecutará?*
 - Utilizar una estructura estandarizada que describa el hardware.
2. *¿Cómo se puede modificar esa información sin que sea necesario reconstruir el kernel?*
 - La estructura no debe estar embebida en el kernel.
 - Se le debe indicar la posición en memoria en la cual se ubica la información a través de un registro estandarizado para cada procesador.

Introducción

Alcance

1. *¿Cómo se le indica al SO sobre que hardware se ejecutará?*
 - Utilizar una estructura estandarizada que describa el hardware.
2. *¿Cómo se puede modificar esa información sin que sea necesario reconstruir el kernel?*
 - La estructura no debe estar embebida en el kernel.
 - Se le debe indicar la posición en memoria en la cual se ubica la información a través de un registro estandarizado para cada procesador.

La estructura, el formato, las herramientas y la estandarización de la metodología de carga componen lo que comúnmente se conoce como *device tree*

Introducción

Ficheros y herramientas

.dtsi

- “**device tree source include**” es un fichero que contiene una representación humanamente legible, en formato texto, del árbol de dispositivos. Por lo general este fichero contiene los dispositivos propios de la plataforma/SoC.

.dts

- “**device tree source**” es un fichero que contiene una representación humanamente legible, en formato texto, del árbol de dispositivos. Por lo general este fichero contiene los dispositivos propios de la placa.

.dtb
(fdt)

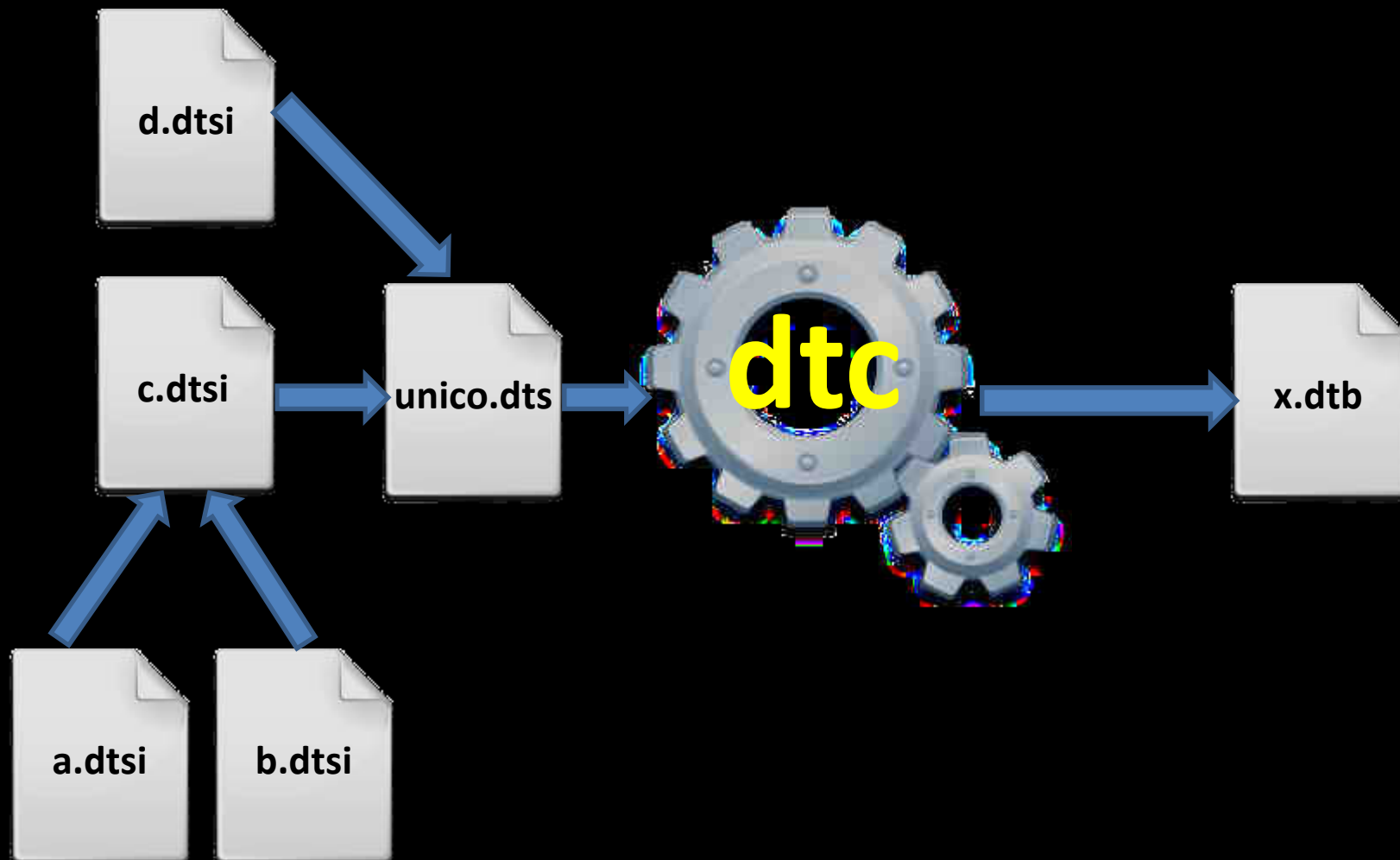
- “**device tree blob**” o “**flatenned device tree**” es fichero binario que contiene una estructura plana utilizada por el programa de carga para transferirle al programa cliente el árbol de dispositivos.

dtc

- “**device tree compiler**” es la herramienta necesaria para generar a partir de los ficheros .dts los .dtb.

Introducción

Ficheros y herramientas



Introducción

Ficheros y herramientas



```
dtc -I dts -O dtb -o /ruta/planta.dtb /ruta/semilla.dts
```



```
dtc -I dtb -O dts -o /ruta/semilla.dts /ruta/planta.dtb
```



```
dtc -I fs -O dts /proc/device-tree/ /ruta/semilla_sistema.dts
```


Árbol De Dispositivos

Introducción

Árbol de dispositivos

- Estructura y convenciones
- Propiedades estándar
- Interrupciones
- Relojes

Nodo de dispositivos

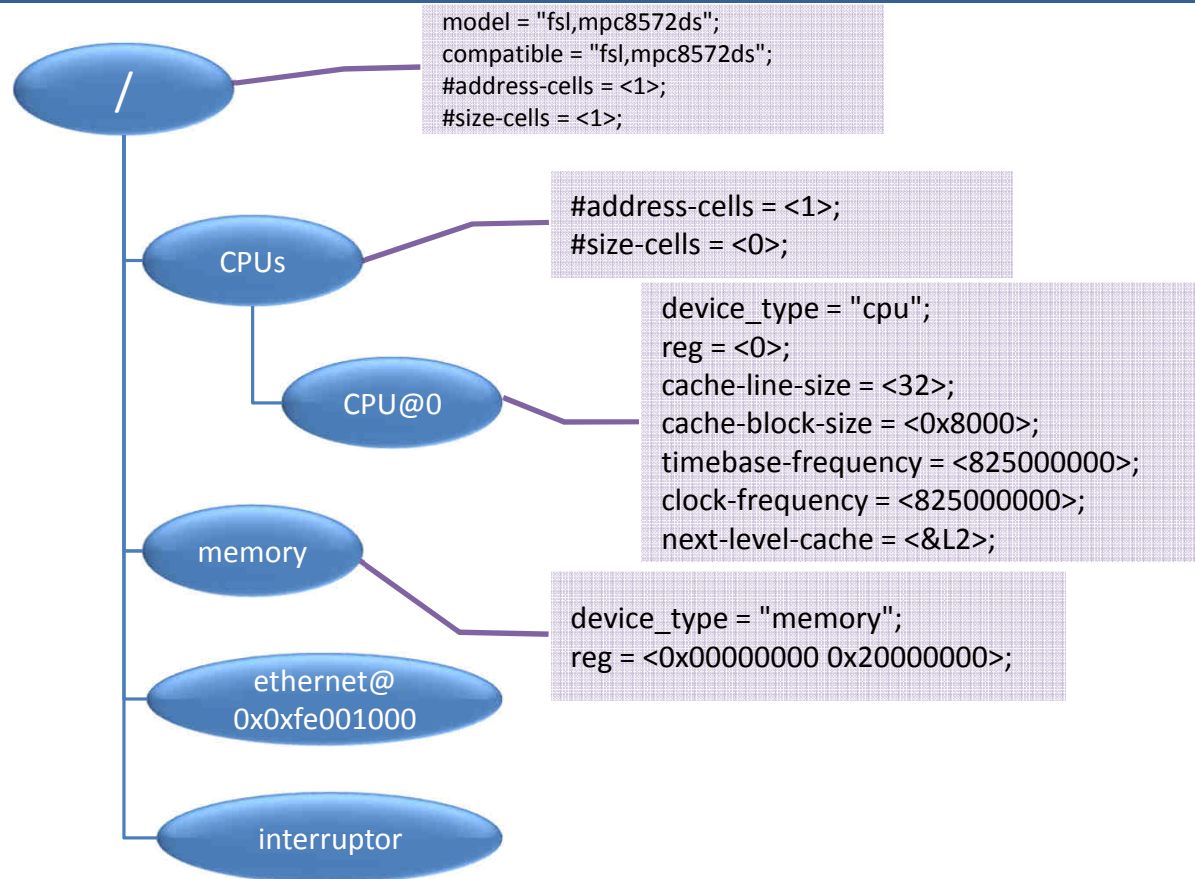
Formato de la imagen

Programa de carga

Casos de estudio

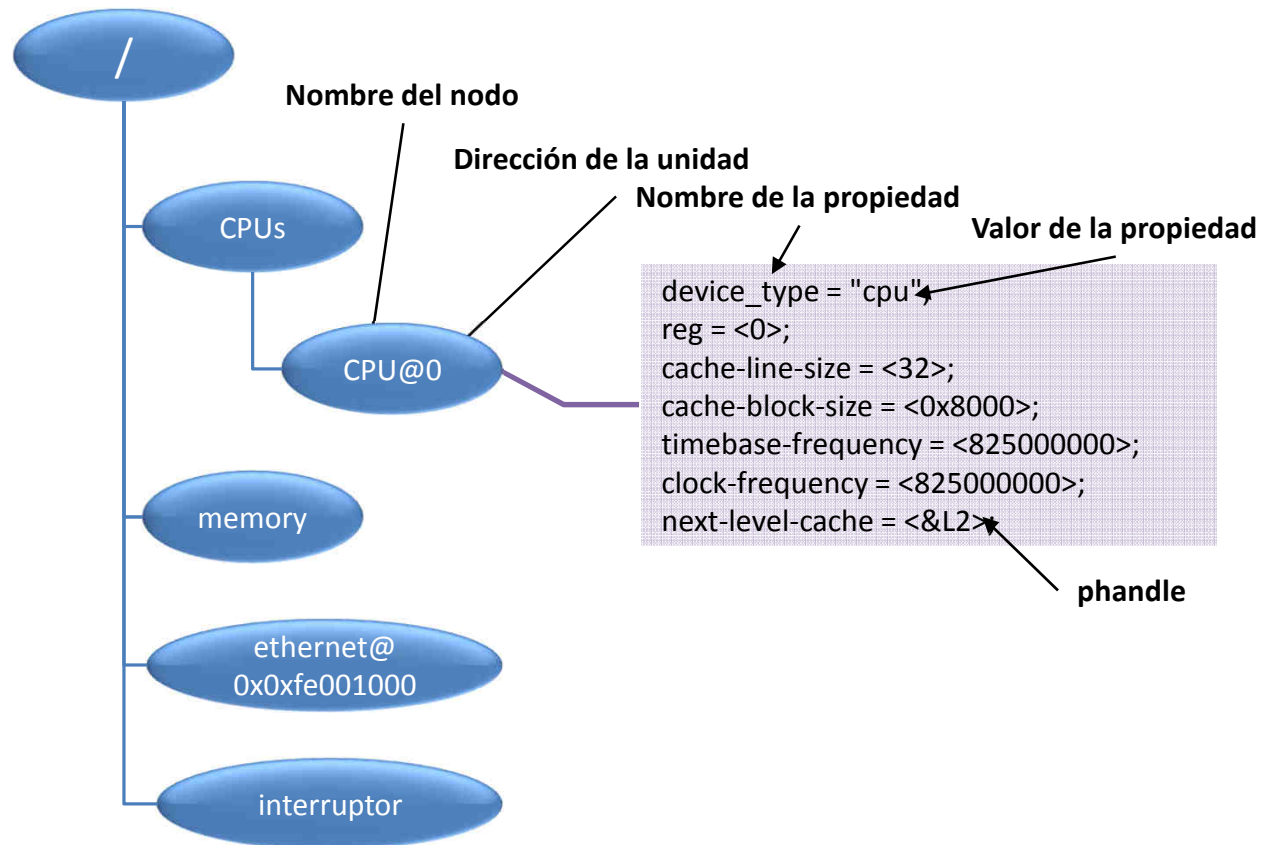
Árbol de dispositivos

Estructura y convenciones



Árbol de dispositivos

Estructura y convenciones



Árbol de dispositivos

Estructura y convenciones

Elementos

Nodo

- **Nombre** Cadena de hasta 31 caracteres utilizada como identificado
- **Dirección** Indica la dirección base del BUS en el cual se encuentra.

Propiedad

- **Nombre** Cadena de hasta 31 caracteres utilizada como identificador
- **Valor** Cadena de bytes en formato big-endian. En caso de ser caracteres se finalizan en NULL por defecto. Las cadenas de caracteres pueden concatenarse mediante " , "

Árbol de dispositivos

Propiedades estándares

compatible

- Lista de cadenas de texto utilizadas para especificar el dispositivo exacto que representa al nodo, el cual será utilizado por el SO para identificar que controlador utilizar, y expresar la compatibilidad con dispositivos similares

model

- Especifica <fabricante,modelo>, su utilización por parte del SO depende del nodo asociado

phandle

- Especifica un identificador numérico único para el nodo del árbol. Este identificador es utilizado para referenciar las propiedades del nodo indicado, por otros nodos del árbol.

status

- Indica el estado del dispositivo ("okay", "disabled", "fail", "fail-sss")

#address-cells/#size-cells

- Indican como debe direccionarse los nodos hijos. Address especifica el numero de direcciones requeridas para acceder al dispositivo. Size el numero de rangos requeridos para especificar el acceso a partir de la dirección base

reg

- Especifica la dirección y rango requerido para acceder al dispositivo dentro el bus. Si el dispositivo dispone de direcciones no consecutivas es posible realizar la concatenación utilizando el separador " ".

virtual-reg

- Permite asociar una dirección virtual a la dirección física especificada en reg

ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados. Utiliza el triplete

dma-ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados, que permitan el acceso a memoria. Utiliza el triplete "child-bus-address,parent-bus-address,length"s

Árbol de dispositivos

Propiedades estándares

compatible

- Lista de cadenas concatenadas utilizadas para expresar la compatibilidad con dispositivos similares

model

- **Especifica <fabricante,modelo>**

phandle

- Especifica un identificador numérico único para el nodo del árbol. Este identificador es utilizado para referenciar las propiedades del nodo indicado, por otros nodos del árbol.

status

- Indica el estado del dispositivo ("okay", "disabled", "fail", "fail-sss")

#address-cells/#size-cells

- Indican como debe direccionarse los nodos hijos. Address especifica el número de direcciones requeridas para acceder al dispositivo. Size el número de rangos requeridos para especificar el acceso a partir de la dirección base

reg

- Especifica la dirección y rango requerido para acceder al dispositivo dentro del bus. Si el dispositivo dispone de direcciones no consecutivas es posible realizar la concatenación utilizando el separador " ".

virtual-reg

- Permite asociar una dirección virtual a la dirección física especificada en reg

ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados. Utiliza el triplete

dma-ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados, que permitan el acceso a memoria. Utiliza el triplete "child-bus-address,parent-bus-address,length"s

Árbol de dispositivos

Propiedades estándares

compatible

- Lista de cadenas concatenadas utilizadas para expresar la compatibilidad con dispositivos similares

model

- Especifica <fabricante,modelo>

phandle

- **Especifica un identificador numérico único para el nodo del árbol. Este identificador es utilizado para referenciar las propiedades del nodo indicado, por otros nodos del árbol.**

status

- Indica el estado del dispositivo ("okay", "disabled", "fail", "fail-sss")

#address-cells/#size-cells

- Indican como debe direccionarse los nodos hijos. Address especifica el número de direcciones requeridas para acceder al dispositivo. Size el número de rangos requeridos para especificar el acceso a partir de la dirección base

reg

- Especifica la dirección y rango requerido para acceder al dispositivo dentro del bus. Si el dispositivo dispone de direcciones no consecutivas es posible realizar la concatenación utilizando el separador " ".

virtual-reg

- Permite asociar una dirección virtual a la dirección física especificada en reg

ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados. Utiliza el triplete

dma-ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados, que permitan el acceso a memoria. Utiliza el triplete "child-bus-address,parent-bus-address,length"s

Árbol de dispositivos

Propiedades estándares

compatible

- Lista de cadenas concatenadas utilizadas para expresar la compatibilidad con dispositivos similares

model

- Especifica <fabricante,modelo>

phandle

- Especifica un identificador numérico único para el nodo del árbol. Este identificador es utilizado para referenciar las propiedades del nodo indicado, por otros nodos del árbol.

status

- **Indica el estado del dispositivo ("okay", "disabled", "fail", "fail-sss")**

#address-cells/#size-cells

- Indican como debe direccionarse los nodos hijos. Address especifica el número de direcciones requeridas para acceder al dispositivo. Size el número de rangos requeridos para especificar el acceso a partir de la dirección base

reg

- Especifica la dirección y rango requerido para acceder al dispositivo dentro del bus. Si el dispositivo dispone de direcciones no consecutivas es posible realizar la concatenación utilizando el separador " ".

virtual-reg

- Permite asociar una dirección virtual a la dirección física especificada en reg

ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados. Utiliza el triplete

dma-ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados, que permitan el acceso a memoria. Utiliza el triplete "child-bus-address,parent-bus-address,length"s *(ing. Christian Nigri)*

Árbol de dispositivos

Propiedades estándares

compatible

- Lista de cadenas concatenadas utilizadas para expresar la compatibilidad con dispositivos similares

model

- Especifica <fabricante,modelo>

phandle

- Especifica un identificador numérico único para el nodo del árbol. Este identificador es utilizado para referenciar las propiedades del nodo indicado, por otros nodos del árbol.

status

- Indica el estado del dispositivo ("okay", "disabled", "fail", "fail-sss")

#address-cells/#size-cells

- **Indican como debe direccionarse los nodos hijos. Address especifica el numero de direcciones requeridas para acceder al dispositivo. Size el numero de rangos requeridos para especificar el acceso a partir de la dirección base**

reg

- Especifica la dirección y rango requerido para acceder al dispositivo dentro del bus. Si el dispositivo dispone de direcciones no consecutivas es posible realizar la concatenación utilizando el separador " ".

virtual-reg

- Permite asociar una dirección virtual a la dirección física especificada en reg

ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados. Utiliza el triplete

dma-ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados, que permitan el acceso a memoria. Utiliza el triplete "child-bus-address,parent-bus-address,length"s

Árbol de dispositivos

Propiedades estándares

compatible

- Lista de cadenas concatenadas utilizadas para expresar la compatibilidad con dispositivos similares

model

- Especifica <fabricante,modelo>

phandle

- Especifica un identificador numérico único para el nodo del árbol. Este identificador es utilizado para referenciar las propiedades del nodo indicado, por otros nodos del árbol.

status

- Indica el estado del dispositivo ("okay", "disabled", "fail", "fail-sss")

#address-cells/#size-cells

- Indican como debe direccionarse los nodos hijos. Address especifica el número de direcciones requeridas para acceder al dispositivo. Size el número de rangos requeridos para especificar el acceso a partir de la dirección base

reg

- **Especifica la dirección y rango requerido para acceder al dispositivo dentro el bus. Si el dispositivo dispone de direcciones no consecutivas es posible realizar la concatenación utilizando el separador " "**.

virtual-reg

- Permite asociar una dirección virtual a la dirección física especificada en reg

ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados. Utiliza el triplete

dma-ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados, que permitan el acceso a memoria. Utiliza el triplete "child-bus-address,parent-bus-address,length"s

Árbol de dispositivos

Propiedades estándares

compatible

- Lista de cadenas concatenadas utilizadas para expresar la compatibilidad con dispositivos similares

model

- Especifica <fabricante,modelo>

phandle

- Especifica un identificador numérico único para el nodo del árbol. Este identificador es utilizado para referenciar las propiedades del nodo indicado, por otros nodos del árbol.

status

- Indica el estado del dispositivo ("okay", "disabled", "fail", "fail-sss")

#address-cells/#size-cells

- Indican como debe direccionarse los nodos hijos. Address especifica el número de direcciones requeridas para acceder al dispositivo. Size el número de rangos requeridos para especificar el acceso a partir de la dirección base

reg

- Especifica la dirección y rango requerido para acceder al dispositivo dentro del bus. Si el dispositivo dispone de direcciones no consecutivas es posible realizar la concatenación utilizando el separador " ".

virtual-reg

- **Permite asociar una dirección virtual a la dirección física especificada en reg**

ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados. Utiliza el triplete

dma-ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados, que permitan el acceso a memoria. Utiliza el triplete "child-bus-address,parent-bus-address,length"s *ing. Cristian Nigri*

Árbol de dispositivos

Propiedades estándares

compatible

- Lista de cadenas concatenadas utilizadas para expresar la compatibilidad con dispositivos similares

model

- Especifica <fabricante,modelo>

phandle

- Especifica un identificador numérico único para el nodo del árbol. Este identificador es utilizado para referenciar las propiedades del nodo indicado, por otros nodos del árbol.

status

- Indica el estado del dispositivo ("okay","disabled","fail","fail-sss")

#address-cells/#size-cells

- Indican como debe direccionarse los nodos hijos. Address especifica el numero de direcciones requeridas para acceder al dispositivo. Size el numero de rangos requeridos para especificar el acceso a partir de la dirección base

reg

- Especifica la dirección y rango requerido para acceder al dispositivo dentro del bus. Si el dispositivo dispone de direcciones no consecutivas es posible realizar la concatenación utilizando el separador " ".

virtual-reg

- Permite asociar una dirección virtual a la dirección física especificada en reg

ranges

- **Provee un método para la traducción del espacio de direcciones físicas de buses relacionados. Utiliza el triplete**

dma-ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados, que permitan el acceso a memoria. Utiliza el triplete "child-bus-address,parent-bus-address,length"s *Ing. Christian Nigri*

Árbol de dispositivos

Propiedades estándares

compatible

- Lista de cadenas concatenadas utilizadas para expresar la compatibilidad con dispositivos similares

model

- Especifica <fabricante,modelo>

phandle

- Especifica un identificador numérico único para el nodo del árbol. Este identificador es utilizado para referenciar las propiedades del nodo indicado, por otros nodos del árbol.

status

- Indica el estado del dispositivo ("okay", "disabled", "fail", "fail-sss")

#address-cells/#size-cells

- Indican como debe direccionarse los nodos hijos. Address especifica el número de direcciones requeridas para acceder al dispositivo. Size el número de rangos requeridos para especificar el acceso a partir de la dirección base

reg

- Especifica la dirección y rango requerido para acceder al dispositivo dentro del bus. Si el dispositivo dispone de direcciones no consecutivas es posible realizar la concatenación utilizando el separador " ".

virtual-reg

- Permite asociar una dirección virtual a la dirección física especificada en reg

ranges

- Provee un método para la traducción del espacio de direcciones físicas de buses relacionados. Utiliza el triplete

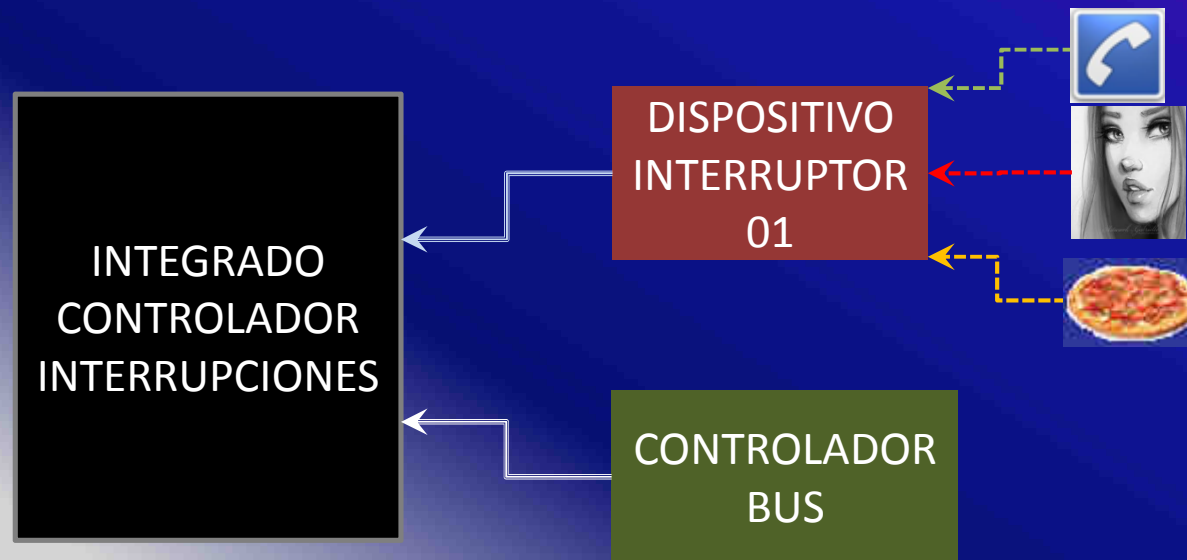
dma-ranges

- **Provee un método para la traducción del espacio de direcciones físicas de buses relacionados, que permitan el acceso a memoria. Utiliza el triplete "child-bus-address,parent-bus-address,length"s**

Árbol de dispositivos

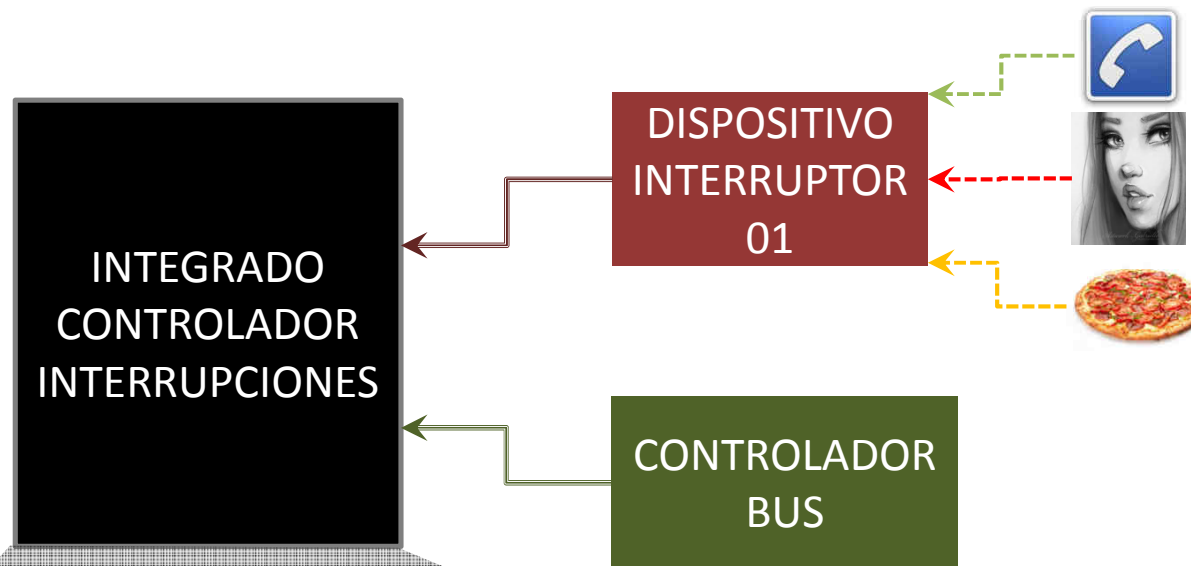
Interrupciones

Hardware



Árbol de dispositivos

Interrupciones

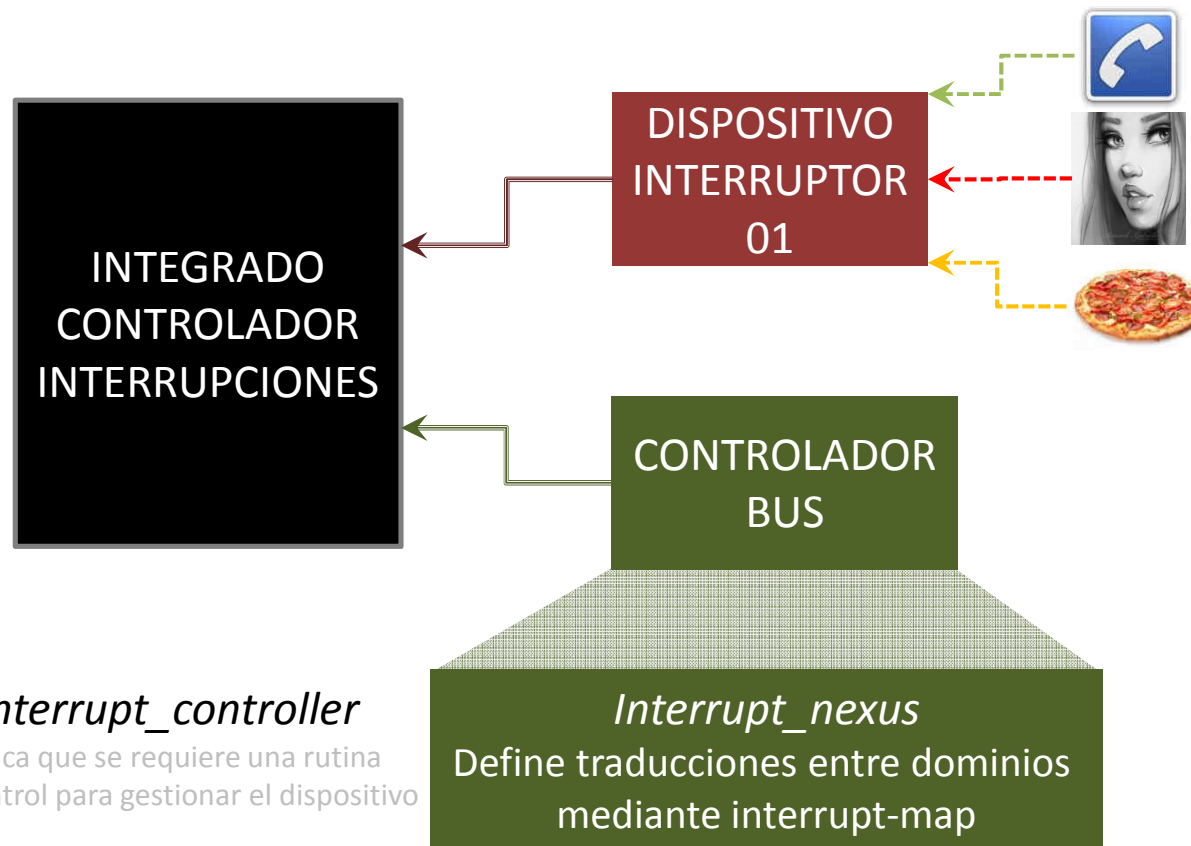


Interrupt_controller

Indica que se requiere una rutina de control para gestionar el dispositivo

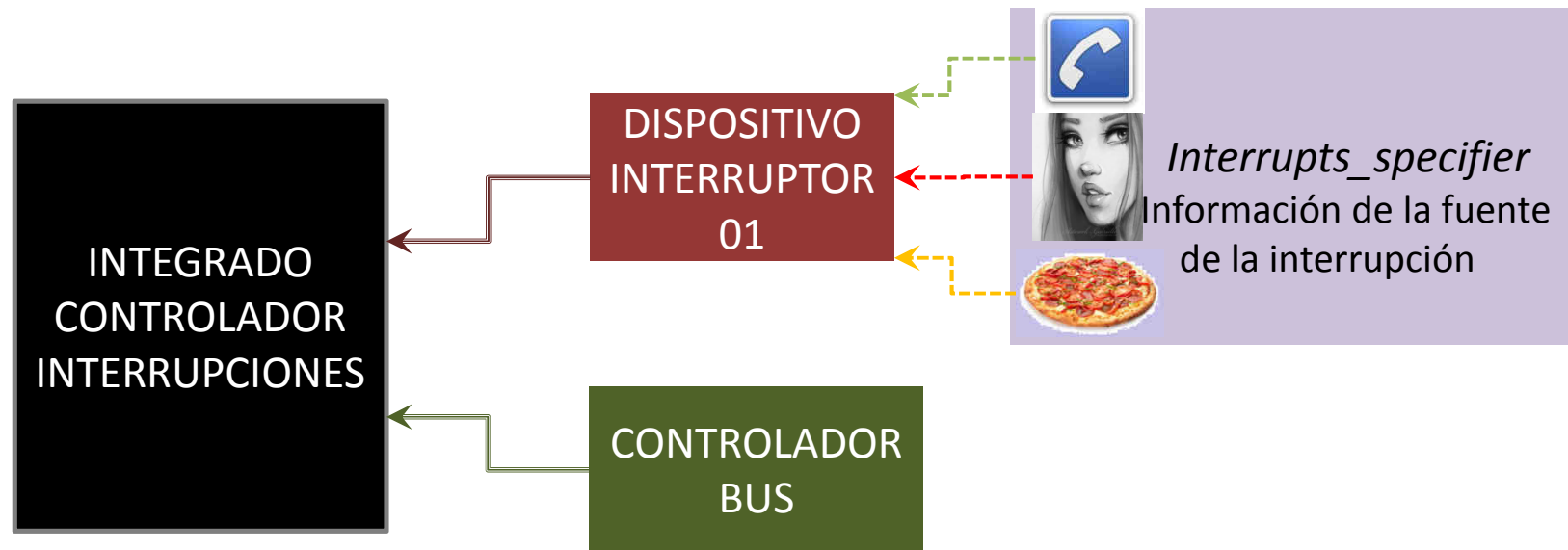
Árbol de dispositivos

Interrupciones



Árbol de dispositivos

Interrupciones



Interrupt_controller

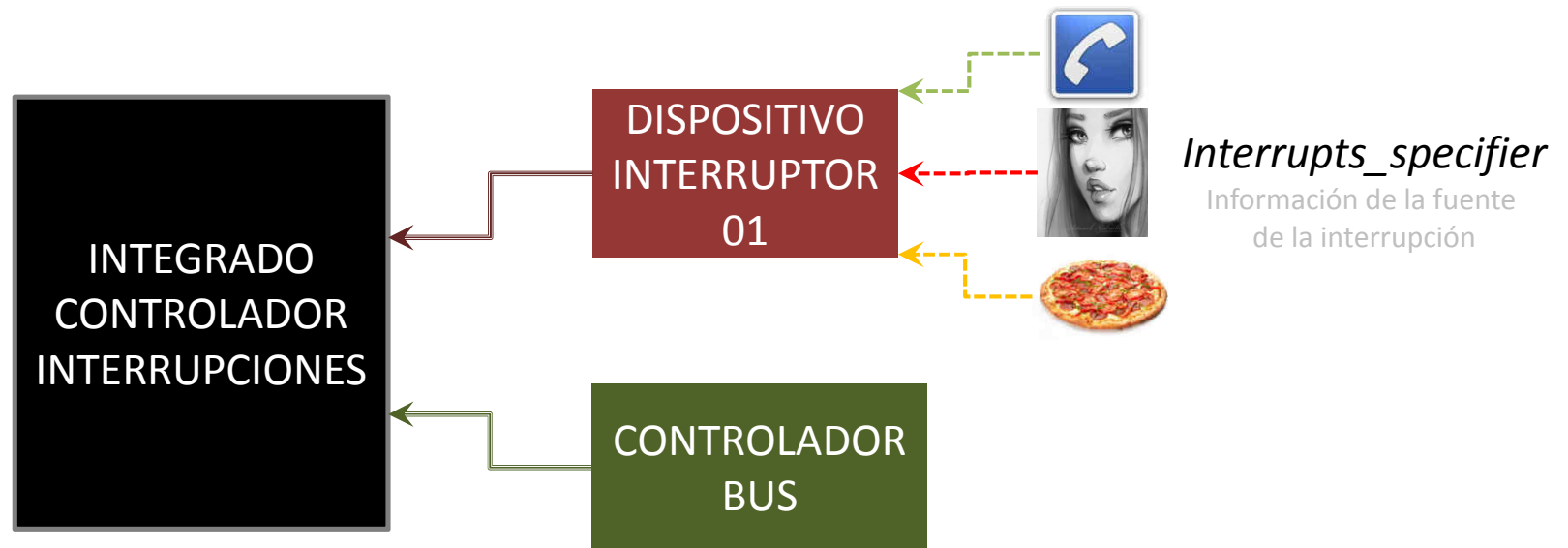
Indica que se requiere una rutina de control para gestionar el dispositivo

Interrupt_nexus

Define traducciones entre dominios mediante interrupt-map

Árbol de dispositivos

Interrupciones



Interrupts_specifier
Información de la fuente de la interrupción

Interrupt_controller

Indica que se requiere una rutina de control para gestionar el dispositivo

Interrupt_nexus

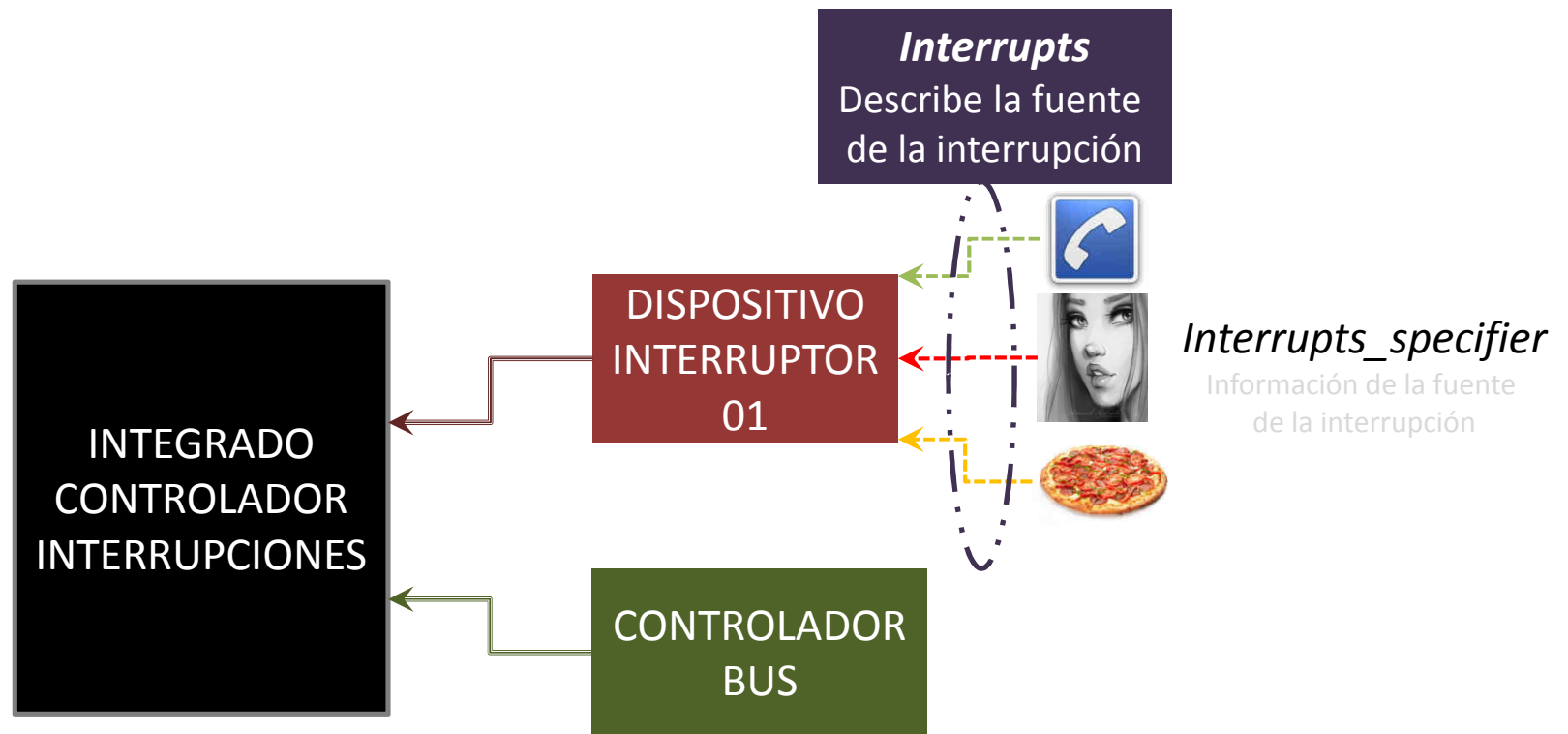
Define traducciones entre dominios mediante interrupt-map

domains

Indica el contexto en el cual un especificador de interrupción es interpretado

Árbol de dispositivos

Interrupciones



Interrupt_controller

Indica que se requiere una rutina de control para gestionar el dispositivo

Interrupt_nexus

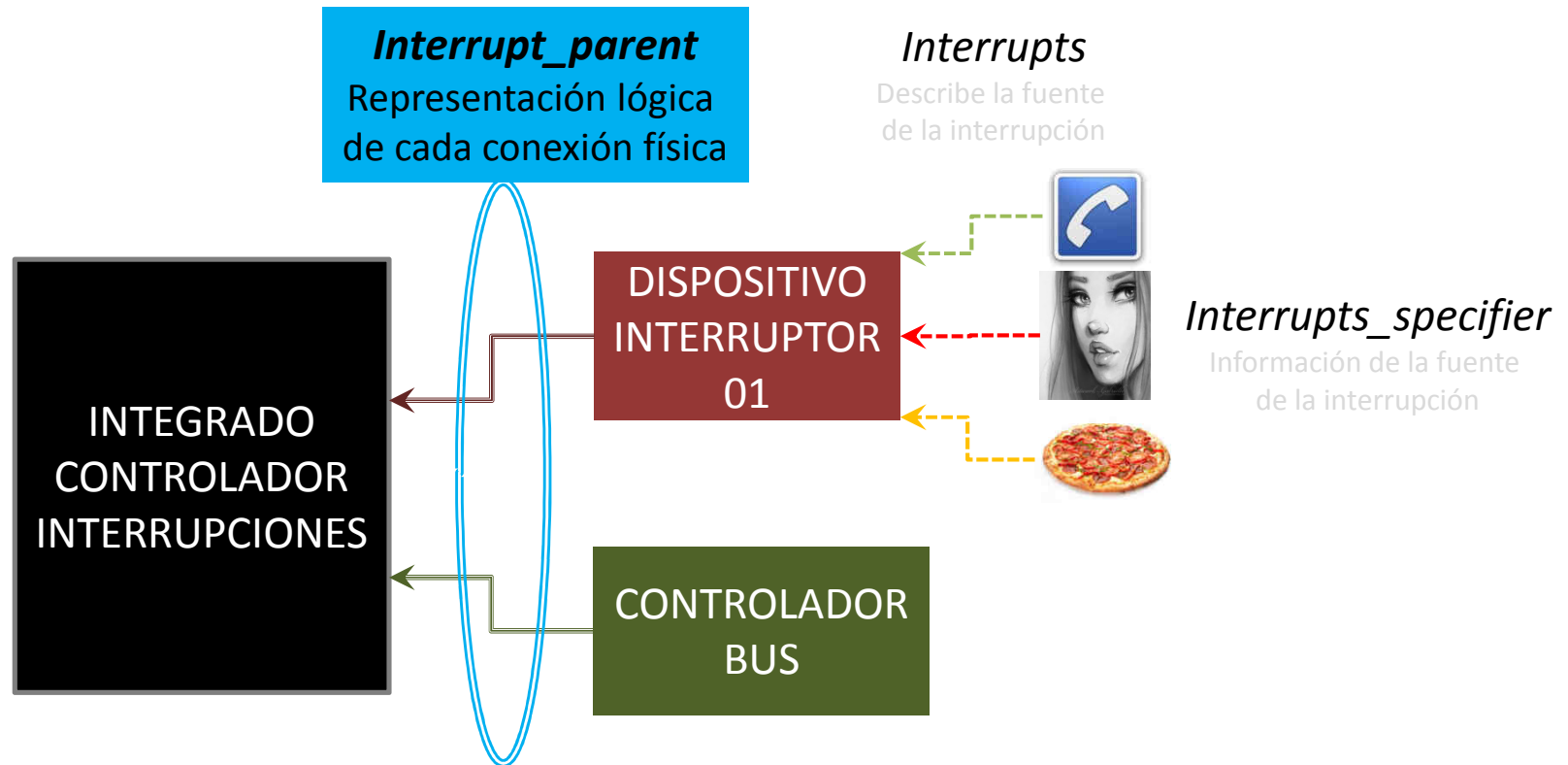
Define traducciones entre dominios mediante interrupt-map

domains

Indica el contexto en el cual un especificador de interrupción es interpretado

Árbol de dispositivos

Interrupciones



Interrupt_controller

Indica que se requiere una rutina de control para gestionar el dispositivo

Interrupt_nexus

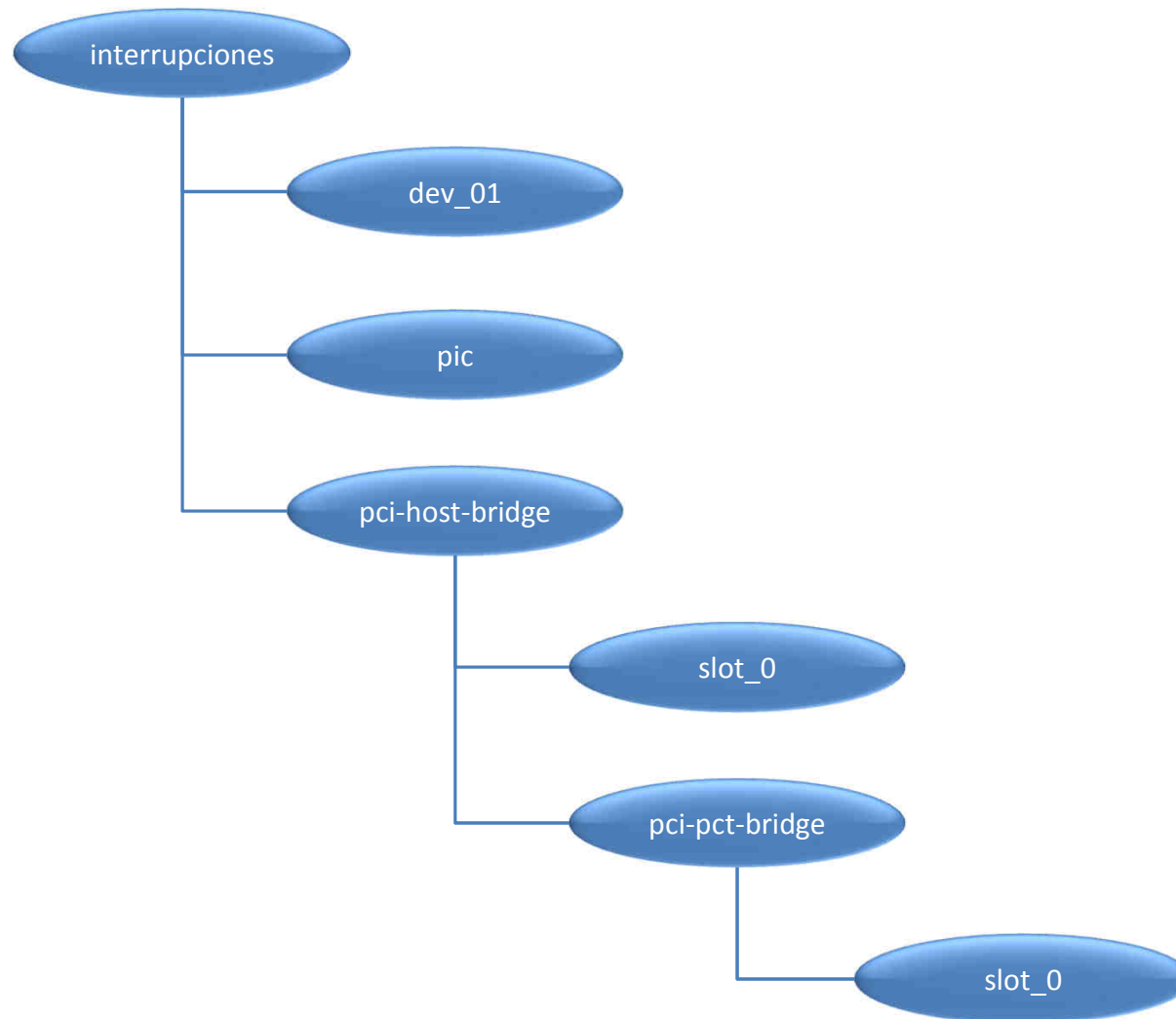
Define traducciones entre dominios mediante interrupt-map

domains

Indica el contexto en el cual un especificador de interrupción es interpretado

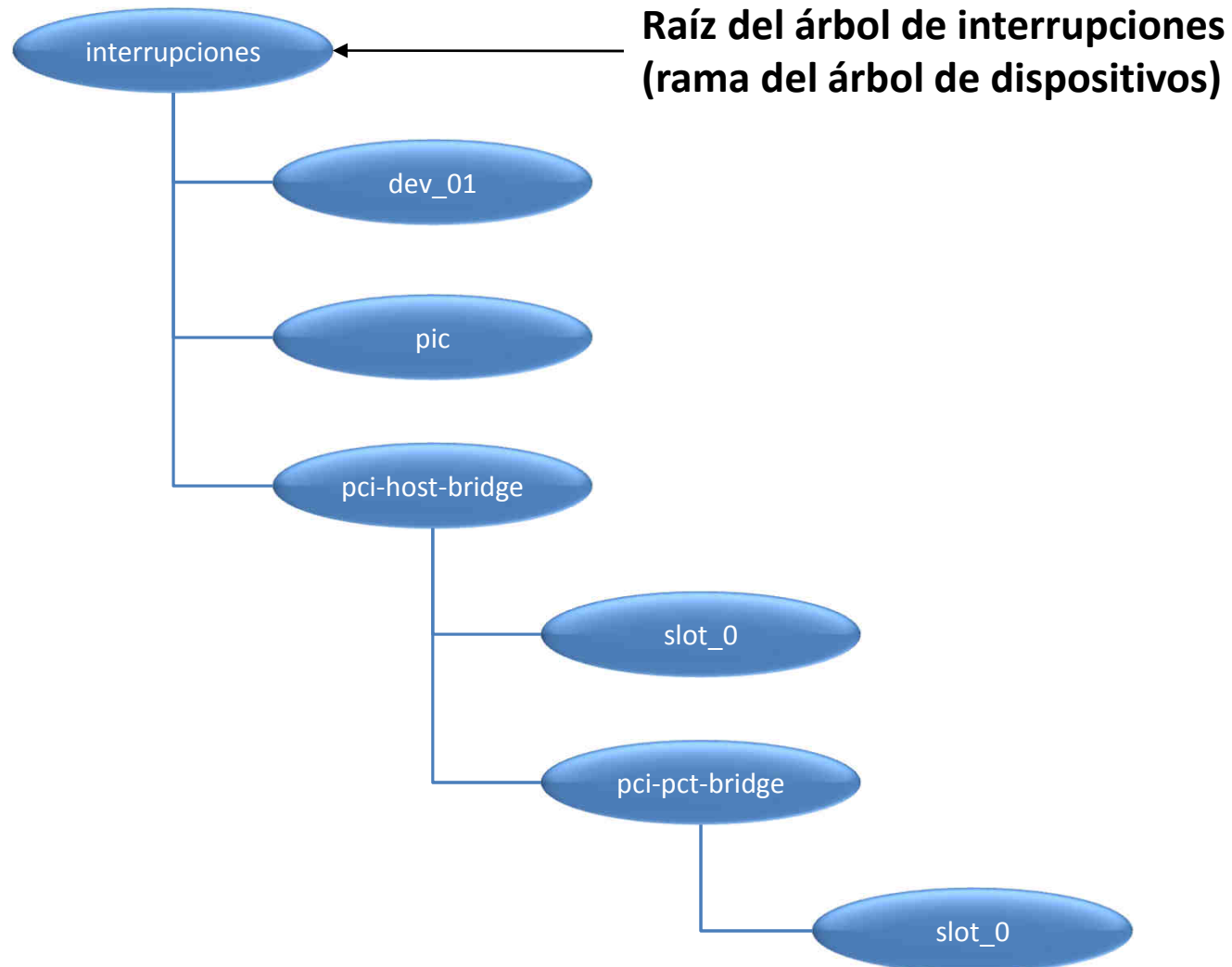
Árbol de dispositivos

Interrupciones



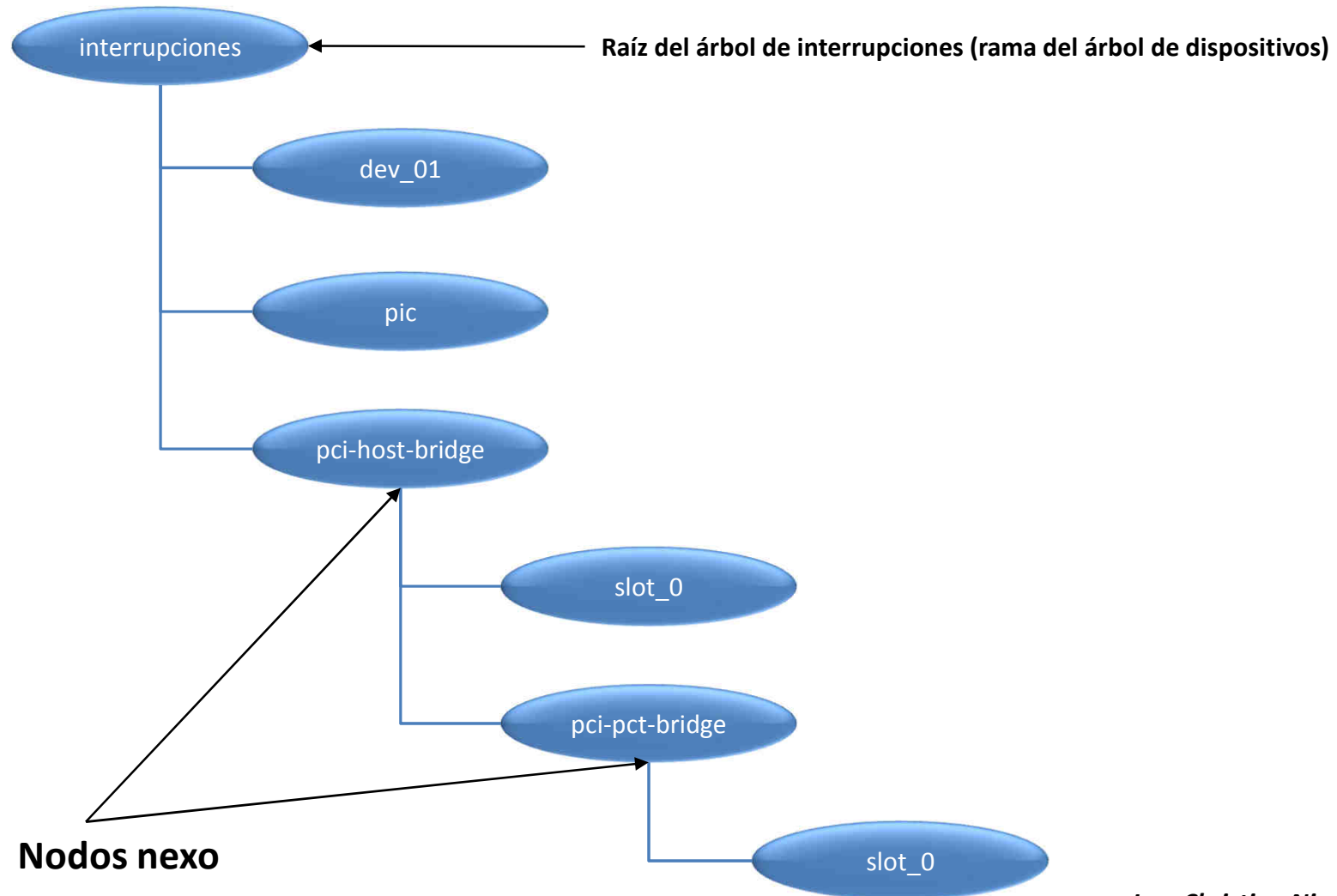
Árbol de dispositivos

Interrupciones



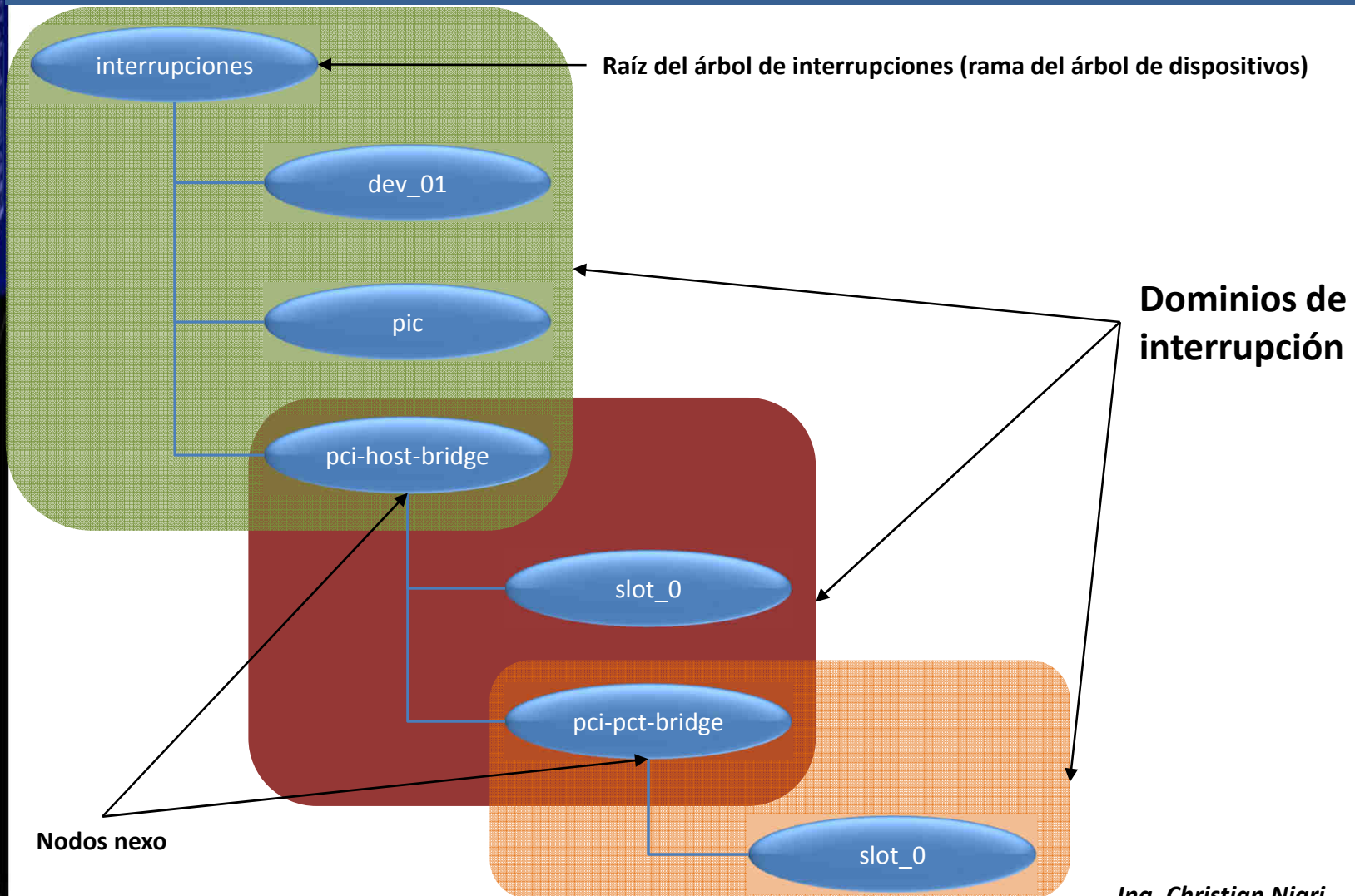
Árbol de dispositivos

Interrupciones



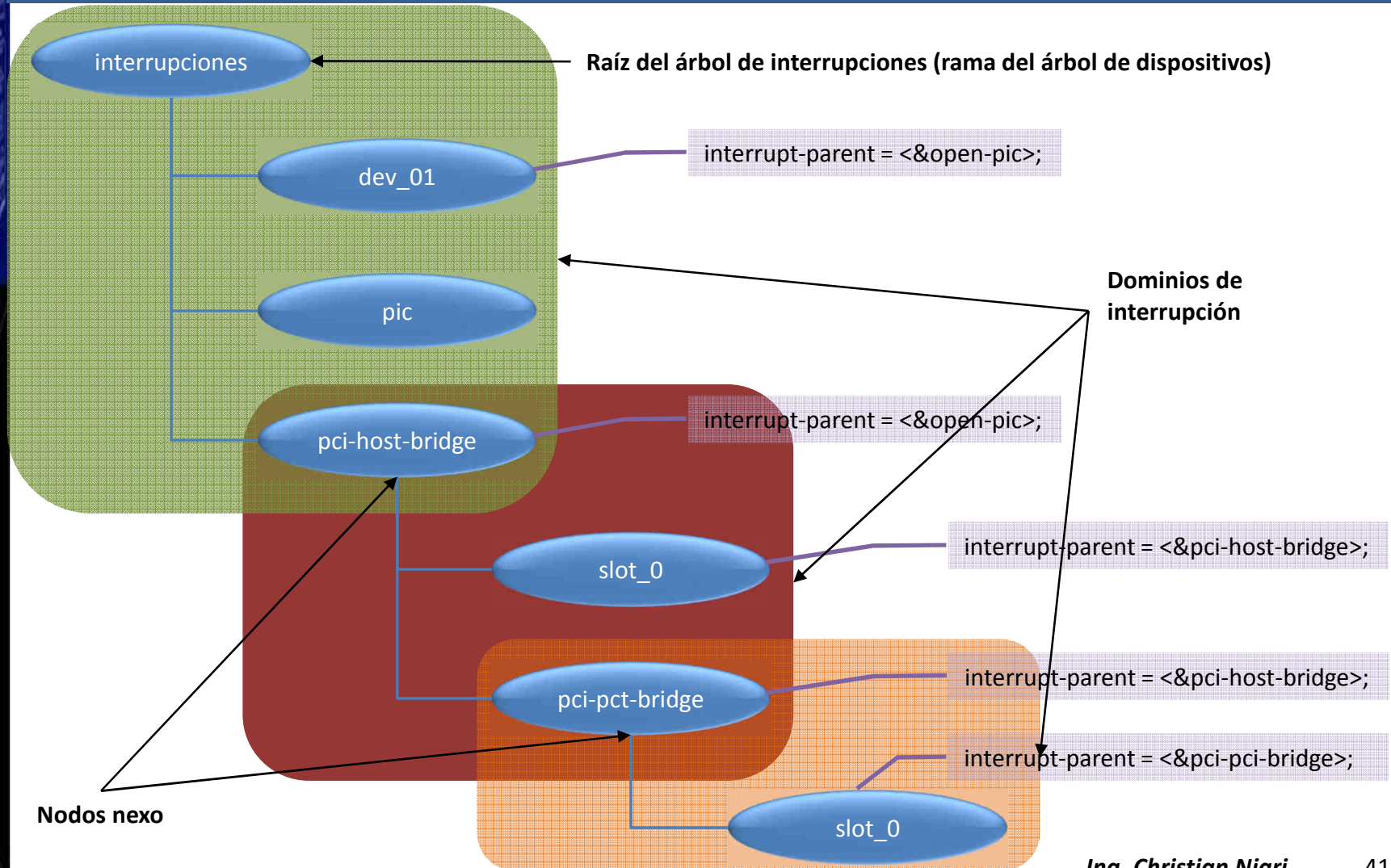
Árbol de dispositivos

Interrupciones



Árbol de dispositivos

Interrupciones





Árbol de dispositivos

Relojes

Ver `/Documentation/devicetree/bindings/clock/clock-bindings.txt`, ojo que es
Un work in progress

Árbol De Dispositivos

Introducción

Árbol de dispositivos

Nodo de dispositivos

Formato de la imagen

Programa de carga

Casos de estudio

Nodos de dispositivos

root

- Solo puede y debe existir un nodo raiz

cpus

- Debe existir un nodo *cpus* que permita agrupar a todos los CPU
 - #address-cells
 - #size-cells

cpu

- Especifica las características del hardware de ejecución
 - device_type
 - Reg
 - clock-frequency
 - timebase-frequency
 - cache-op-block-size
 - reservation-granule-size
 - Status
 - enable-method
 - cpu-release-addr
 - power-isa-version
 - power-isa-*
 - mmu-type

memory

- Debe existir al menos un nodo del tipo memoria
 - device_type
 - reg
 - initial-mapped-area

aliases

- Permite indicar las propiedades de un nodo a través de alias a otro nodo

chosen

- Permite seleccionar los parámetros a utilizar por un nodo en tiempo de ejecución.

Nodos de dispositivos

root

- Solo puede y debe existir un nodo raiz

cpus

- Debe existir un nodo *cpus* que permita agrupar a todos los CPU
 - #address-cells
 - #size-cells

cpu

- Especifica las características del hardware de ejecución
 - device_type
 - Reg
 - clock-frequency
 - timebase-frequency
 - cache-op-block-size
 - reservation-granule-size
 - Status
 - enable-method
 - cpu-release-addr
 - power-isa-version
 - power-isa-*
 - mmu-type

memory

- Debe existir al menos un nodo del tipo memoria
 - device_type
 - reg
 - initial-mapped-area

aliases

- Permite indicar las propiedades de un nodo a través de alias a otro nodo

chosen

- Permite seleccionar los parámetros a utilizar por un nodo en tiempo de ejecución.

Nodos de dispositivos

root

- Solo puede y debe existir un nodo raiz

cpus

- Debe existir un nodo *cpus* que permita agrupar a todos los CPU
 - #address-cells
 - #size-cells

cpu

- Especifica las características del hardware de ejecución
 - device_type
 - Reg
 - clock-frequency
 - timebase-frequency
 - cache-op-block-size
 - reservation-granule-size
 - Status
 - enable-method
 - cpu-release-addr
 - power-isa-version
 - power-isa-*
 - mmu-type

memory

- Debe existir al menos un nodo del tipo memoria
 - device_type
 - reg
 - initial-mapped-area

aliases

- Permite indicar las propiedades de un nodo a través de alias a otro nodo

chosen

- Permite seleccionar los parámetros a utilizar por un nodo en tiempo de ejecución.

Nodos de dispositivos

root

- Solo puede y debe existir un nodo raiz

cpus

- Debe existir un nodo *cpus* que permita agrupar a todos los CPU
 - #address-cells
 - #size-cells

cpu

- Especifica las características del hardware de ejecución
 - device_type
 - Reg
 - clock-frequency
 - timebase-frequency
 - cache-op-block-size
 - reservation-granule-size
 - Status
 - enable-method
 - cpu-release-addr
 - power-isa-version
 - power-isa-*
 - mmu-type

memory

- Debe existir al menos un nodo del tipo memoria
 - device_type
 - reg
 - initial-mapped-area

aliases

- Permite indicar las propiedades de un nodo a través de alias a otro nodo

chosen

- Permite seleccionar los parámetros a utilizar por un nodo en tiempo de ejecución.

Nodos de dispositivos

root

- Solo puede y debe existir un nodo raiz

cpus

- Debe existir un nodo *cpus* que permita agrupar a todos los CPU
 - #address-cells
 - #size-cells

cpu

- Especifica las características del hardware de ejecución
 - device_type
 - Reg
 - clock-frequency
 - timebase-frequency
 - cache-op-block-size
 - reservation-granule-size
 - Status
 - enable-method
 - cpu-release-addr
 - power-isa-version
 - power-isa-*
 - mmu-type

memory

- Debe existir al menos un nodo del tipo memoria
 - device_type
 - reg
 - initial-mapped-area

aliases

- Permite indicar las propiedades de un nodo a través de alias a otro nodo

chosen

- Permite seleccionar los parámetros a utilizar por un nodo en tiempo de ejecución.

Nodos de dispositivos

root

- Solo puede y debe existir un nodo raiz

cpus

- Debe existir un nodo *cpus* que permita agrupar a todos los CPU
 - #address-cells
 - #size-cells

cpu

- Especifica las características del hardware de ejecución
 - device_type
 - Reg
 - clock-frequency
 - timebase-frequency
 - cache-op-block-size
 - reservation-granule-size
 - Status
 - enable-method
 - cpu-release-addr
 - power-isa-version
 - power-isa-*
 - mmu-type

memory

- Debe existir al menos un nodo del tipo memoria
 - device_type
 - reg
 - initial-mapped-area

aliases

- Permite indicar las propiedades de un nodo a través de alias a otro nodo

chosen

- Permite seleccionar los parámetros a utilizar por un nodo en tiempo de ejecución.

Árbol De Dispositivos

Introducción

Árbol de dispositivos

Nodo de dispositivos

Formato de la imagen

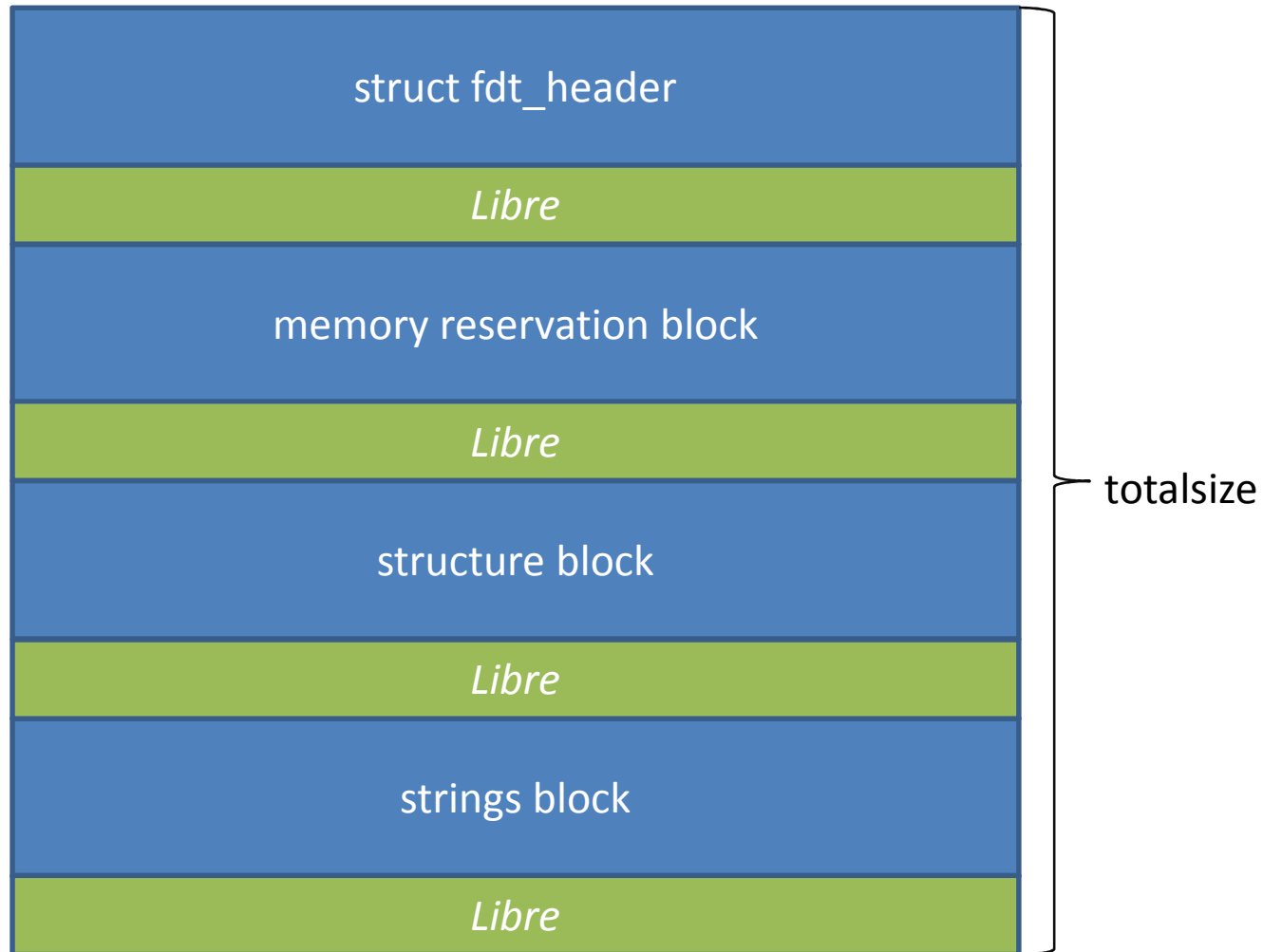
- FDT/DTB
- FIT

Programa de carga

Casos de estudio

Formato de la imagen

Flattened Device Tree /Device Tree Blob



Formato de la imagen

FDT/DTB

struct fdt_header

```
struct fdt_header {  
    uint32_t magic;  
    /*0xd00dfeed */  
    uint32_t totalsize;  
    uint32_t off_dt_struct;  
    uint32_t off_dt_strings;  
    uint32_t off_mem_rsvmap;  
    uint32_t version;  
    uint32_t last_comp_version;  
    uint32_t boot_cpuid_phys;  
    uint32_t size_dt_strings;  
    uint32_t size_dt_struct;  
};
```

Formato de la imagen

FDT/DTB

struct fdt_header

```
struct fdt_header {  
    uint32_t magic;  
    uint32_t totalsize;  
    uint32_t off_dt_struct;  
    uint32_t off_dt_strings;  
    uint32_t off_mem_rsvmap;  
    uint32_t version;  
    uint32_t last_comp_version;  
    uint32_t boot_cpuid_phys;  
    uint32_t size_dt_strings;  
    uint32_t size_dt_struct;  
};
```

Formato de la imagen

FDT/DTB

struct fdt_header

```
struct fdt_header {
    uint32_t magic;
    uint32_t totalsize;
    uint32_t off_dt_struct;
    /*desplazamiento, en bytes, respecto al
    inicio de la cabecera del "structure
    block"*/
    uint32_t off_dt_strings;
    uint32_t off_mem_rsvmap;
    uint32_t version;
    uint32_t last_comp_version;
    uint32_t boot_cpuid_phys;
    uint32_t size_dt_strings;
    uint32_t size_dt_struct;
};
```

Formato de la imagen

FDT/DTB

struct fdt_header

```
struct fdt_header {
    uint32_t magic;
    uint32_t totalsize;
    uint32_t off_dt_struct;
    uint32_t off_dt_strings;
    /*desplazamiento, en bytes, respecto al
inicio de la cabecera del “strings
block”*/
    uint32_t off_mem_rsvmap;
    uint32_t version;
    uint32_t last_comp_version;
    uint32_t boot_cpuid_phys;
    uint32_t size_dt_strings;
    uint32_t size_dt_struct;
};
```

Formato de la imagen

FDT/DTB

struct fdt_header

```
struct fdt_header {  
    uint32_t magic;  
    uint32_t totalsize;  
    uint32_t off_dt_struct;  
    uint32_t off_dt_strings;  
    uint32_t off_mem_rsvmap;  
    /*desplazamiento, en bytes, respecto al  
    inicio de la cabecera del "memory  
    reservation block"*/  
    uint32_t version;  
    uint32_t last_comp_version;  
    uint32_t boot_cpuid_phys;  
    uint32_t size_dt_strings;  
    uint32_t size_dt_struct;  
};
```


Formato de la imagen

FDT/DTB

struct fdt_header

```
struct fdt_header {  
    uint32_t magic;  
    uint32_t totalsize;  
    uint32_t off_dt_struct;  
    uint32_t off_dt_strings;  
    uint32_t off_mem_rsvmap;  
    uint32_t version;  
    /*Actualmente 17 (feb/16)*/  
    uint32_t last_comp_version;  
    uint32_t boot_cpuid_phys;  
    uint32_t size_dt_strings;  
    uint32_t size_dt_struct;  
};
```

Formato de la imagen

FDT/DTB

struct fdt_header

```
struct fdt_header {  
    uint32_t magic;  
    uint32_t totalsize;  
    uint32_t off_dt_struct;  
    uint32_t off_dt_strings;  
    uint32_t off_mem_rsvmap;  
    uint32_t version;  
    uint32_t last_comp_version;  
    /*versión compatible mas antigua*/  
    uint32_t boot_cpuid_phys;  
    uint32_t size_dt_strings;  
    uint32_t size_dt_struct;  
};
```

Formato de la imagen

FDT/DTB

struct fdt_header

```
struct fdt_header {
    uint32_t magic;
    uint32_t totalsize;
    uint32_t off_dt_struct;
    uint32_t off_dt_strings;
    uint32_t off_mem_rsvmap;
    uint32_t version;
    uint32_t last_comp_version;
    uint32_t boot_cpuid_phys;
    /*Identificador del CPU utilizado para
    inicializar el sistema. Debe coincidir
    con el indicado en la propiedad reg el
    nodo CPU del .dts*/
    uint32_t size_dt_strings;
    uint32_t size_dt_struct;
};
```

Formato de la imagen

FDT/DTB

struct fdt_header

```
struct fdt_header {  
    uint32_t magic;  
    uint32_t totalsize;  
    uint32_t off_dt_struct;  
    uint32_t off_dt_strings;  
    uint32_t off_mem_rsvmap;  
    uint32_t version;  
    uint32_t last_comp_version;  
    uint32_t boot_cpuid_phys;  
    uint32_t size_dt_strings;  
    /*Tamaño en bytes de la sección "strings  
    block"*/  
    uint32_t size_dt_struct;  
};
```

Formato de la imagen

FDT/DTB

struct fdt_header

```
struct fdt_header {  
    uint32_t magic;  
    uint32_t totalsize;  
    uint32_t off_dt_struct;  
    uint32_t off_dt_strings;  
    uint32_t off_mem_rsvmap;  
    uint32_t version;  
    uint32_t last_comp_version;  
    uint32_t boot_cpuid_phys;  
    uint32_t size_dt_strings;  
    uint32_t size_dt_struct;  
    /*Tamaño en bytes de la sección  
    “stuct block”*/  
};
```

Formato de la imagen

FDT/DTB



The diagram shows a rectangular box divided into two horizontal sections. The top section is blue and contains the text 'struct fdt_header'. The bottom section is green and contains the text 'Libre'.

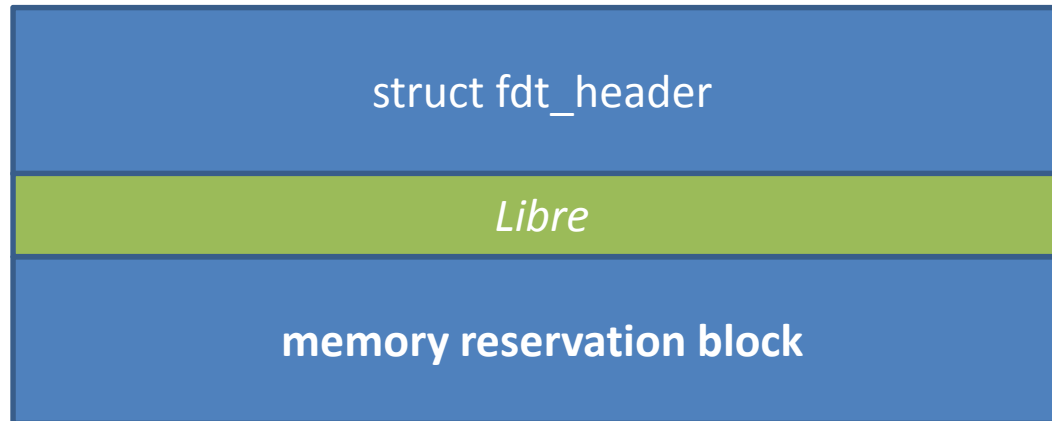
```
struct fdt_header
```

Libre

Las secciones libres solo debe utilizarse para preservar la lineación de las secciones ubicadas a continuación de las mismas.
Las secciones deben alinearse en límites de 8 bytes.

Formato de la imagen

FDT/DTB



Le indica al programa cliente las zonas de la memoria física que se encuentran reservadas.

```
/*
```

```
Máximo 64 zonas
```

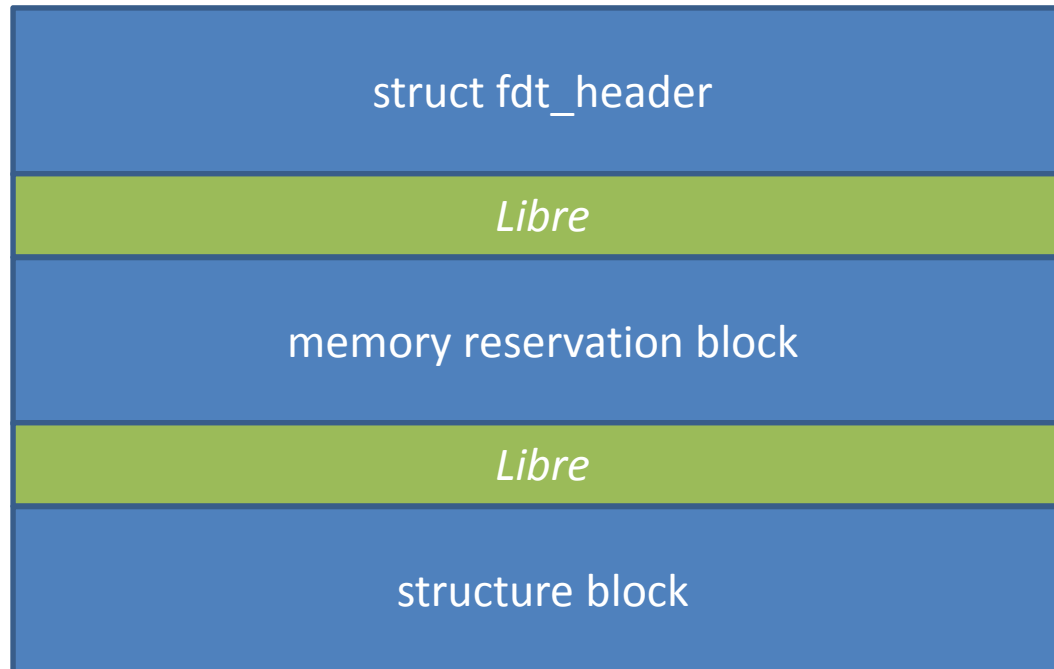
```
Finaliza con una estructura inicializada en 0
```

```
*/
```

```
struct fdt_reserve_entry {  
    uint64_t address;  
    uint64_t size;  
};
```

Formato de la imagen

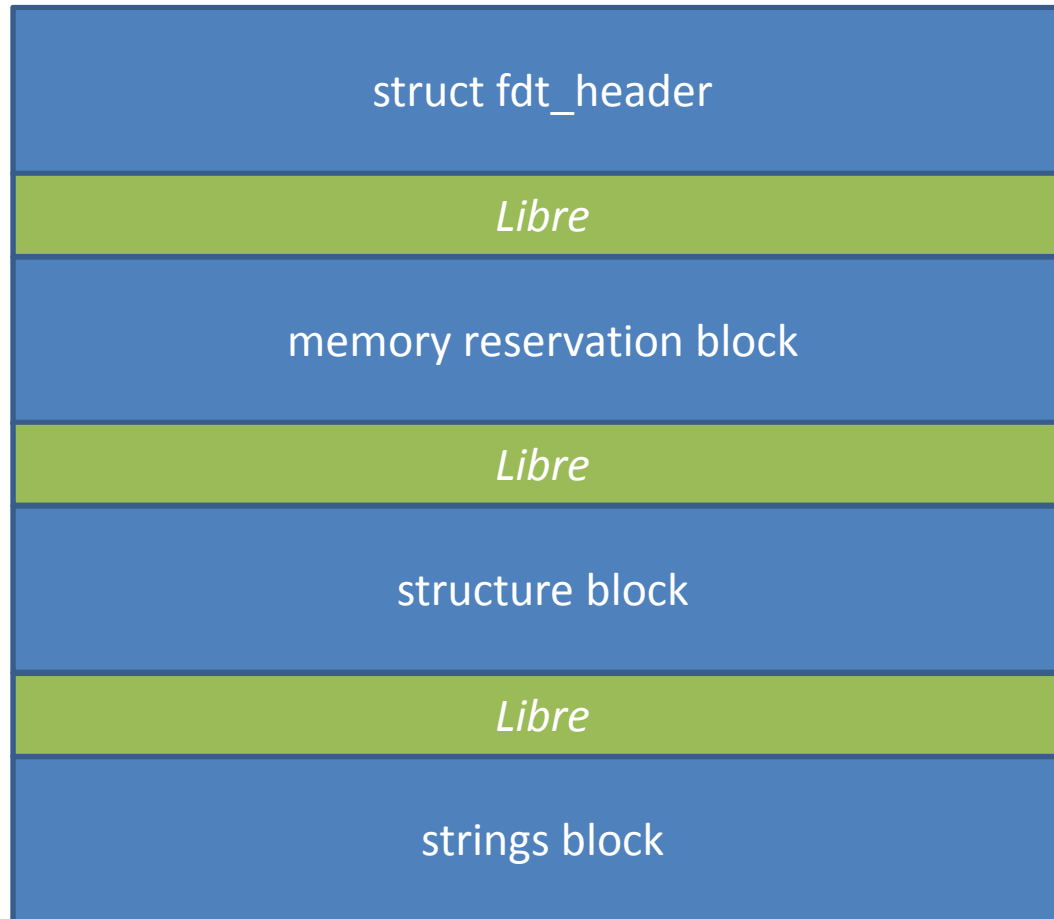
FDT/DTB



Contiene el .dts compilado.

Formato de la imagen

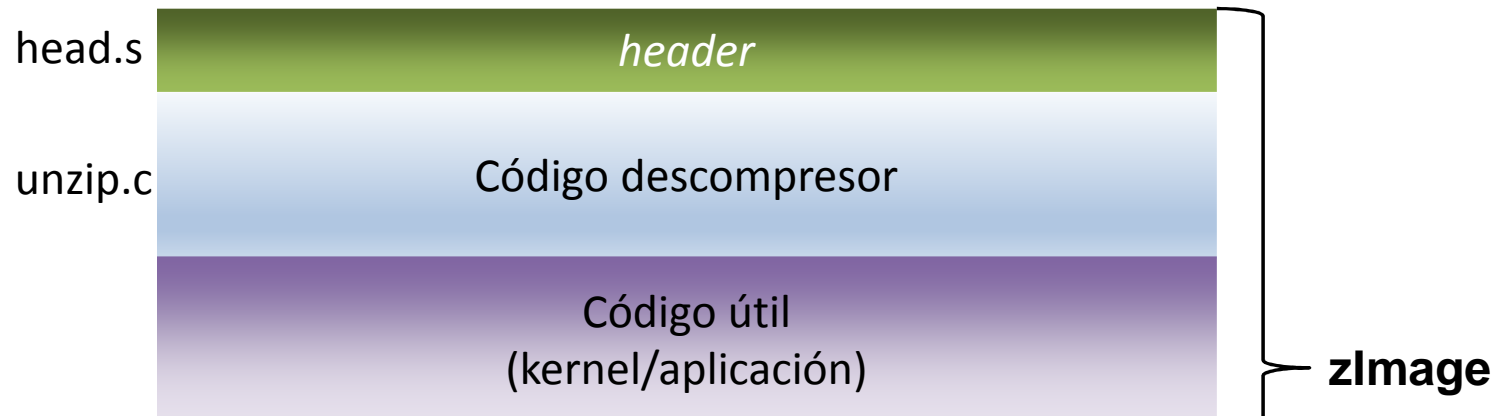
FDT/DTB



Contiene las cadenas de caracteres terminadas en NULL que representan el nombre de las propiedades utilizadas en los arboles. Las cadenas se concatenan en forma simple.

Formato de la imagen

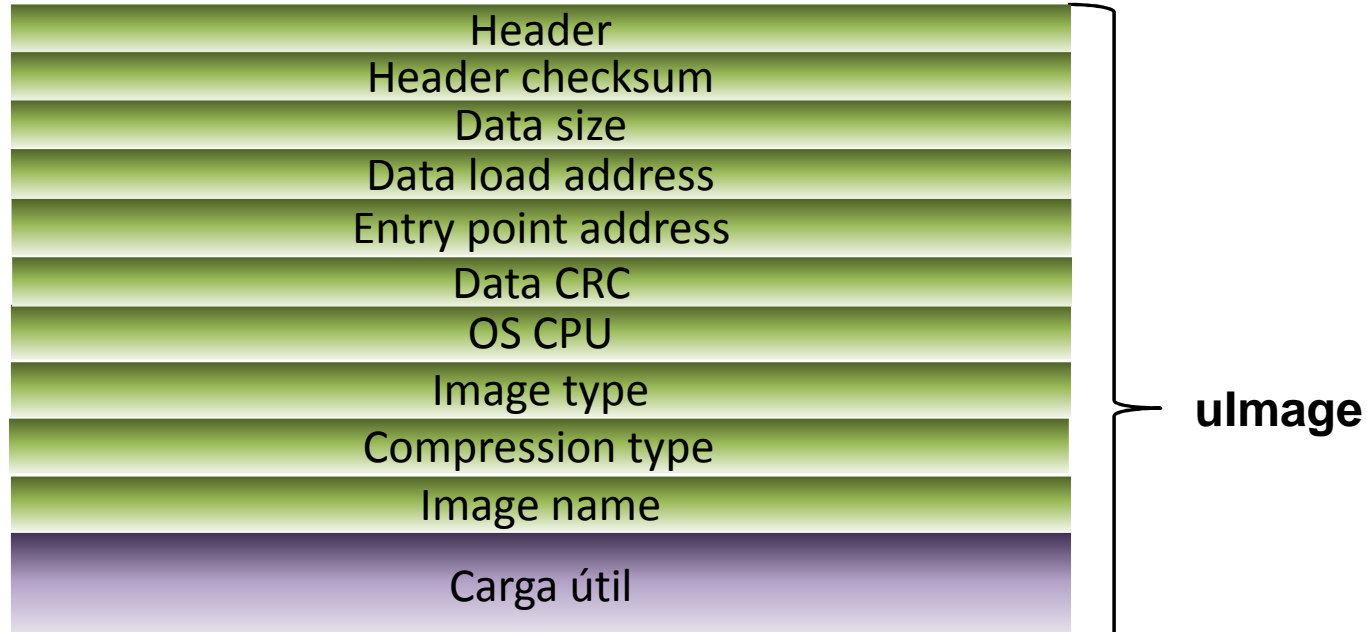
Flattened Image Tree



- ↓ No dispone de seguridad
- ↓ No se verifica la integridad del “checksum”
- ↓ No se verifica la integridad del “hash”
- ↓ No permite múltiples imágenes

Formato de la imagen

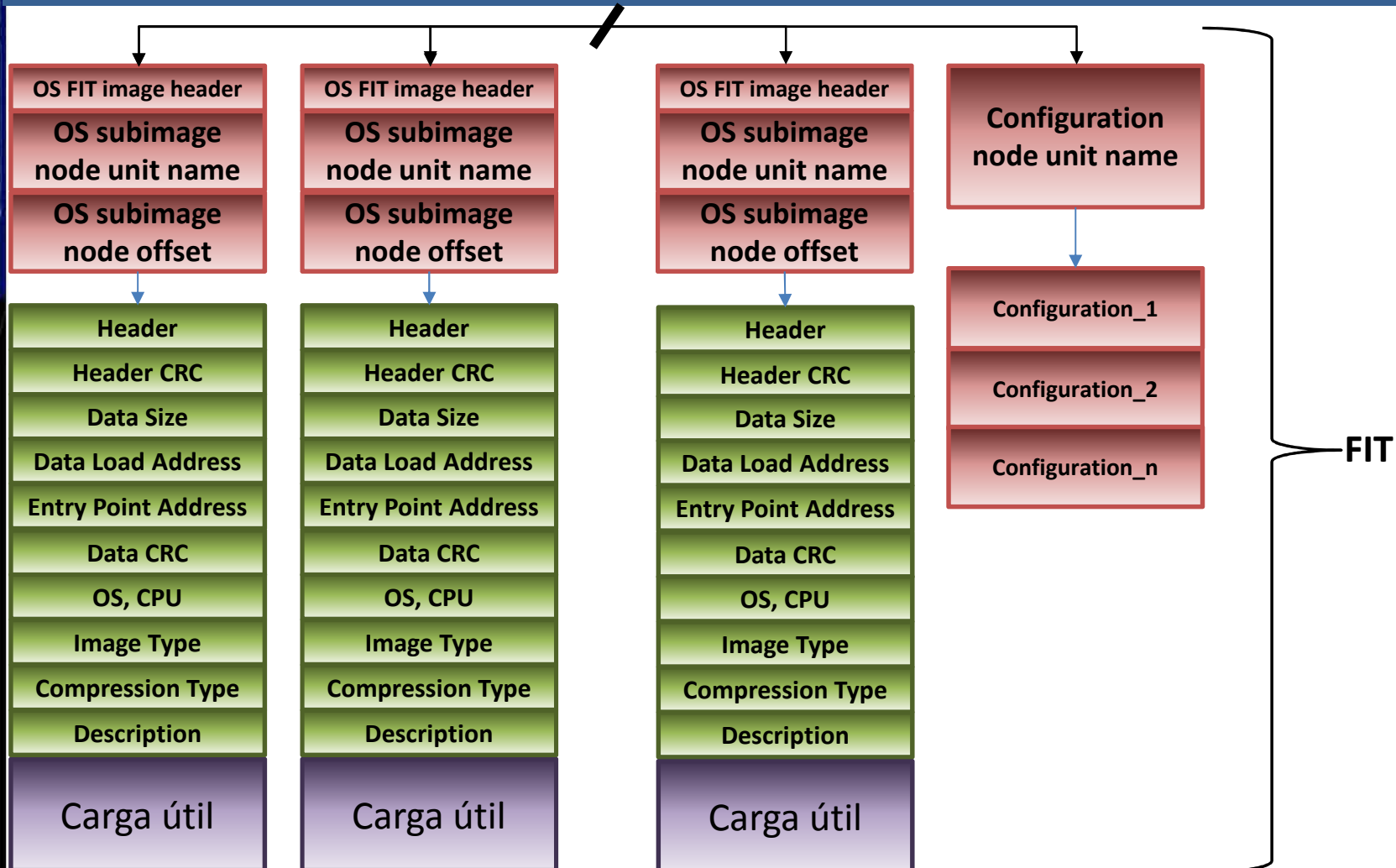
FIT



- ↓ No dispone de seguridad
- ↓ No se verifica la integridad del "hash"
- ↓ No permite múltiples imágenes

Formato de la imagen

FIT



Formato de la imagen

FIT

```

/dts-v1/;
/{
  description = "Christimage con Linux kernel, FDT y ramdisk"
  #address-cells = <0x1>;

  configurations {
    default = "conf@1";
    conf@1 {
      description = "Kernel-FDT-Ramdisk"
      kernel = "kernel@1";
      fdt = "fdt@1";
      ramdisk = "ramdisk@1"
    };
  };

  images {
    kernel@1 {
      description = "Chris Linux kernel"
      data = /incbin*("./vmlinux.bin.gz");
      type = "kernel";
      arch = "arm";
      os = "linux";
      compression = "gzip";
      load = <0xA000>;
      entry = <0xA000>;
      hash@1 {
        algo = "md5";
      };
      hash@2 {
        algo = "sha256";
      };
    };

    ramdisk@1 {
      description = "Ramdisk Image"
      data = /incbin*("./ramdisk.images.gz");
      type = "ramdisk";
      arch = "arm";
      os = "linux";
      compression = "gzip";
      load = <0x00A00000>;
      entry = <0x00A00000>;
      hash@1 {
        algo = "md5";
      };
      hash@2 {
        algo = "sha256";
      };
    };

    fdt@1 {
      description = "Chris FDT"
      data = /incbin*("./boby.dtb");
      type = "flat_dt";
      arch = "arm";
      compression = "none";
      hash@1 {
        algo = "md5";
      };
      hash@2 {
        algo = "sha1";
      };
    };
  };
};

```

Árbol De Dispositivos

Introducción

Árbol de dispositivos

Nodo de dispositivos

Formato de la imagen

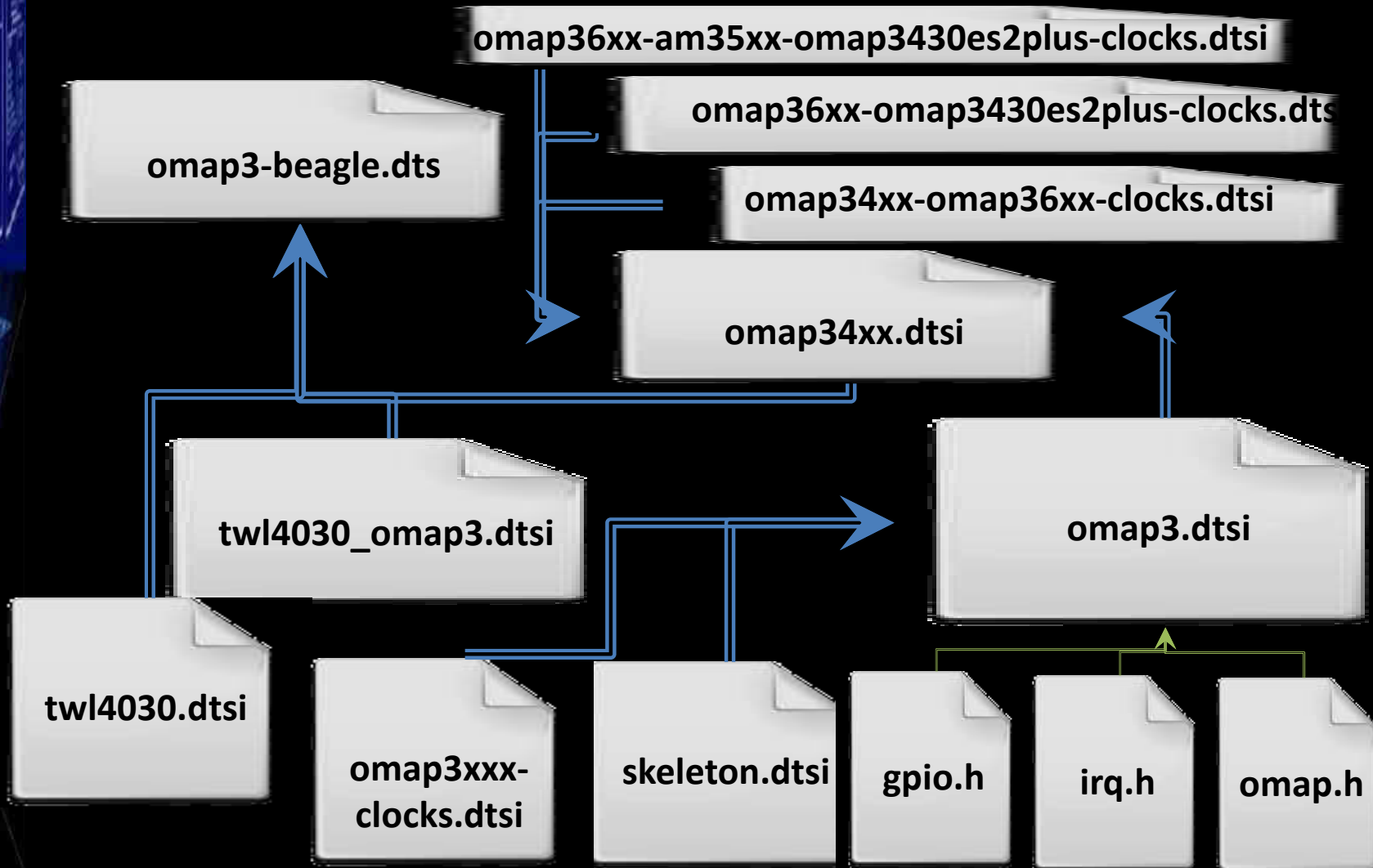
Programa de carga

Casos de estudio

- Beagle
- Galileo
- iMX25

Casos de estudio

Beagle





Casos de estudio

Beagle

```
1 /*
2  * Copyright (C) 2012 Texas Instruments Incorporated - http://www.ti.com/
3  * This program is free software; you can redistribute it and/or modify
4  * it under the terms of the GNU General Public License version 2 as
5  * published by the Free Software Foundation.
6  */
7
8 /dts-v1/;
9
10 #include "omap34xx.dtsi"
11
12 / {
13     model = "TI OMAP3 BeagleBoard";
14     compatible = "ti,omap3-beagle", "ti,omap3";
15
16     cpus {
17         cpu@0 {
18             cpu0-supply = <&vcc>;
19         };
20     };
21
22     memory {
23         device_type = "memory";
24         reg = <0x80000000 0x10000000>; /* 256 MB */
25     };
26
27     aliases {
28         display0 = &dvi0;
29         display1 = &tv0;
30     };
31 }
```

omap3-beagle.dts

Casos de estudio

Beagle

```
196     uart3_pins: pinmux_uart3_pins {
197         pinctrl-single,pins = <
198             0x16e (PIN_INPUT | PIN_OFF_WAKEUPENABLE | MUX_MODE0)
199             /* uart3_rx_irrx.uart3_rx_irrx */
200             0x170 (PIN_OUTPUT | MUX_MODE0) /* uart3_tx_irtx.uart3_tx_irtx */
201         >;
202     };

318 &uart3 {
319     pinctrl-names = "default";
320     pinctrl-0 = <&uart3_pins>;
321     interrupts-extended = <&intc 74 &omap3_pmx_core OMAP3_UART3_RX>;
322 };
```

omap3-beagle.dts

Casos de estudio

Beagle

```
196     uart3_pins: pinmux_uart3_pins {
197         pinctrl-single,pins = <
198             0x16e (PIN_INPUT | PIN_OFF_WAKEUPENABLE | MUX_MODE0)
199             /* uart3_rx_irrx.uart3_rx_irrx */
200             0x170 (PIN_OUTPUT | MUX_MODE0)
201             /* uart3_tx_irtx.uart3_tx_irtx */
202         >;
203     };
```

→ BINDING: La funcionalidad de *pinconf* no esta soportada

omap3-beagle.dts

Casos de estudio

Beagle

```
196     uart3_pins: pinmux_uart3_pins {
197         pinctrl-single,pins = <
198             0x16e (PIN_INPUT | PIN_OFF_WAKEUPENABLE | MUX_MODE0)
199             /* uart3_rx_irrx.uart3_rx_irrx */
200             0x170 (PIN_OUTPUT | MUX_MODE0)
201             /* uart3_tx_irtx.uart3_tx_irtx */
202         >;
203     };
```

PROPERTY : Solo los valores especificados en el BINDING `pinctrl-single, function-mask` serán actualizados

omap3-beagle.dts

Casos de estudio

Beagle

```
196     uart3_pins: pinmux_uart3_pins {
197         pinctrl-single,pins = <
198             0x16e (PIN_INPUT | PIN_OFF_WAKEUPENABLE | MUX_MODE0)
199             /* uart3_rx_irrx.uart3_rx_irrx */
200             0x170 (PIN_OUTPUT | MUX_MODE0)
201             /* uart3_tx_irtx.uart3_tx_irtx */
202         >;
203     };
```

→ REG: desplazamiento respecto de la dirección base de los registros específicos del IO/MUX

VALUE: valor aplicable (dependiente de la mascara)

omap3-beagle.dts



Casos de estudio

Beagle

```
1 /*
2  * Device Tree Source for OMAP3 SoC
3  *
4  * Copyright (C) 2011 Texas Instruments Incorporated - http://www.ti.com/
5  *
6  * This file is licensed under the terms of the GNU General Public License
7  * version 2. This program is licensed "as is" without any warranty of any
8  * kind, whether express or implied.
9  */
10
11 #include <dt-bindings/gpio/gpio.h>
12 #include <dt-bindings/interrupt-controller/irq.h>
13 #include <dt-bindings/pinctrl/omap.h>
14
15 #include "skeleton.dtsi"
16
17 / {
18     compatible = "ti,omap3430", "ti,omap3";
19     interrupt-parent = <&intc>;
20
21     aliases {
22         i2c0 = &i2c1;
23         i2c1 = &i2c2;
24         i2c2 = &i2c3;
25         serial0 = &uart1;
26         serial1 = &uart2;
27         serial2 = &uart3;
28     };
```

omap3.dtsi

Casos de estudio

Beagle

```
74      /*
75      * XXX: Use a flat representation of the OMAP3 interconnect.
76      * The real OMAP interconnect network is quite complex.
77      * Since it will not bring real advantage to represent that in DT for
78      * the moment, just use a fake OCP bus entry to represent the whole bus
79      * hierarchy.
80      */
81      ocp {
82          compatible = "ti,omap3-l3-smx", "simple-bus";
83          reg = <0x68000000 0x10000>;
84          interrupts = <9 10>;
85          #address-cells = <1>;
86          #size-cells = <1>;
87          ranges;
88          ti,hwmods = "l3_main";
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104      uart3: serial@49020000 {
105          compatible = "ti,omap3-uart";
106          reg = <0x49020000 0x400>;
107          interrupts-extended = <&intc 74>;
108          dmas = <&sdma 53 &sdma 54>;
109          dma-names = "tx", "rx";
110          ti,hwmods = "uart3";
111          clock-frequency = <48000000>;
112      };
```

Open Core Protocol: Standard para la integración de componentes virtuales (OCP-IP)

Ing. Christian Nigri

omap3.dtsi

Casos de estudio

Beagle

```
304     uart3: serial@49020000 {
305         compatible = "ti,omap3-uart";
306         reg = <0x49020000 0x400>;
307         interrupts-extended = <&intc 74>;
308         dmas = <&sdma 53 &sdma 54>;
309         dma-names = "tx", "rx";
310         ti,hwmods = "uart3";
311         clock-frequency = <48000000>;
312     };
```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

omap3.dtsi

Casos de estudio

Beagle

```
304         uart3: serial@49020000 {
305             compatible = "ti,omap3-uart";
306             reg = <0x49020000 0x400>;
307             interrupts-extended = <&intc 74>;
308             dmas = <&sdma 53 &sdma 54>;
309             dma-names = "tx", "rx";
310             ti,hwmods = "uart3";
311             clock-frequency = <48000000>;
312         };
```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

→ PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

omap3.dtsi

Casos de estudio

Beagle

```
304         uart3: serial@49020000 {
305             compatible = "ti,omap3-uart";
306             reg = <0x49020000 0x400>;
307             interrupts-extended = <&intc 74>;
308             dmas = <&sdma 53 &sdma 54>;
309             dma-names = "tx", "rx";
310             ti,hwmods = "uart3";
311             clock-frequency = <48000000>;
312         };
```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

→ PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

omap3.dtsi

Casos de estudio

Beagle

```
304         uart3: serial@49020000 {
305             compatible = "ti,omap3-uart";
306             reg = <0x49020000 0x400>;
307             interrupts-extended = <&intc 74>;
308             dmas = <&sdma 53 &sdma 54>;
309             dma-names = "tx", "rx";
310             ti,hwmods = "uart3";
311             clock-frequency = <48000000>;
312         };
```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

PROPERTY: Configuración por defecto del dispositivo

→ PROPERTY: IRQ asociada al dispositivo

omap3.dtsi

Casos de estudio

Beagle

```
304         uart3: serial@49020000 {
305             compatible = "ti,omap3-uart";
306             reg = <0x49020000 0x400>;
307             interrupts-extended = <&intc 74>;
308             dmas = <&sdma 53 &sdma 54>;
309             dma-names = "tx", "rx";
310             ti,hwmods = "uart3";
311             clock-frequency = <48000000>;
312         };
```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

PROPERTY: IRQ asociada al dispositivo

→ PROPERTY: phandle al nodo DMA, los canales a utilizar y el *friendly-name*

omap3.dtsi

Casos de estudio

Beagle

```
304     uart3: serial@49020000 {
305         compatible = "ti,omap3-uart";
306         reg = <0x49020000 0x400>;
307         interrupts-extended = <&intc 74>;
308         dmas = <&sdma 53 &sdma 54>;
309         dma-names = "tx", "rx";
310         ti,hwmods = "uart3";
311         clock-frequency = <48000000>;
312     };
```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

PROPERTY: IRQ asociada al dispositivo

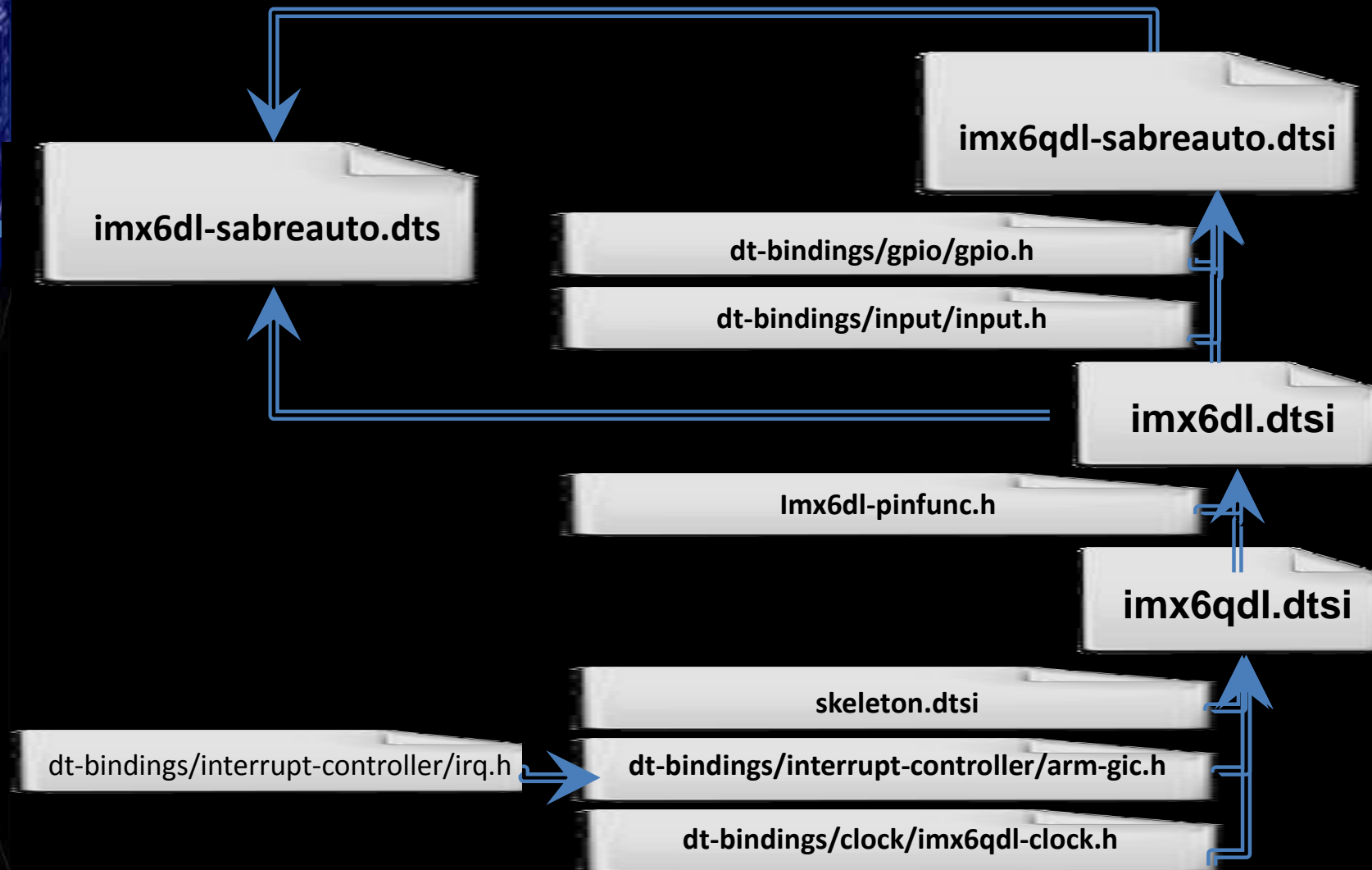
PROPERTY: phandle al nodo DMA, los canales a utilizar y el *friendly-name*

PROPERTY: asocia el dispositivo con un bloque IP específico del fabricante (*omap_hwmod.c*)

omap3.dtsi

Casos de estudio

iMX6-Sabreauto



Casos de estudio

Bindings

```
542.     gpio1: gpio@0209c000 {
543.         compatible = "fsl,imx6q-gpio", "fsl,imx35-gpio";
544.         reg = <0x0209c000 0x4000>;
545.         interrupts = <0 66 IRQ_TYPE_LEVEL_HIGH>,
546.                    <0 67 IRQ_TYPE_LEVEL_HIGH>;
547.         gpio-controller;
548.         #gpio-cells = <2>;
549.         interrupt-controller;
550.         #interrupt-cells = <2>;
551.     };
```

imx6qdl.dtsi



Casos de estudio

Bindings

```
542.     gpio1: gpio@0209c000 {
543.         compatible = "fsl,imx6q-gpio", "fsl,imx35-gpio";
544.         reg = <0x0209c000 0x4000>;
545.         interrupts = <0 66 IRQ_TYPE_LEVEL_HIGH>,
546.                   <0 67 IRQ_TYPE_LEVEL_HIGH>;
547.         gpio-controller;
548.         #gpio-cells = <2>;
549.         interrupt-controller;
550.         #interrupt-cells = <2>;
551.     };
```

→ LABEL: NODE NAME : Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

imx6qdl.dtsi

Casos de estudio

Bindings

```
542.     gpio1: gpio@0209c000 {
543.         compatible = "fsl,imx6q-gpio", "fsl,imx35-gpio";
544.         reg = <0x0209c000 0x4000>;
545.         interrupts = <0 66 IRQ_TYPE_LEVEL_HIGH>,
546.                    <0 67 IRQ_TYPE_LEVEL_HIGH>;
547.         gpio-controller;
548.         #gpio-cells = <2>;
549.         interrupt-controller;
550.         #interrupt-cells = <2>;
551.     };
```

imx6qd1.dtsi

```
1128.     &usdhc1 { pinctrl-names = "default";
1129.               pinctrl-0 = <&pinctrl_usdhc1>;
1130.               cd-gpios = <&gpio1 1 GPIO_ACTIVE_LOW>;
1131.               no-1-8-v;
1132.               keep-power-in-suspend;
1133.               enable-sdio-wakeup;
1134.               status = "okay";
1135.     };
```

imx6qd1-sabreauto.dtsi

Casos de estudio

Bindings

```

542. gpio1: gpio@0209c000 {
543.     compatible = "fsl,imx6q-gpio", "fsl,imx35-gpio";
544.     reg = <0x0209c000 0x4000>;
545.     interrupts = <0 66 IRQ_TYPE_LEVEL_HIGH,
546.                 <0 67 IRQ_TYPE_LEVEL_HIGH>;
547.     gpio-controller;
548.     #gpio-cells = <2>;
549.     interrupt-controller;
550.     #interrupt-cells = <2>;
551. };

```

```

663. &iomuxc { pinctrl-names = "default";
664.         pinctrl-0 = <&pinctrl_hog>;
665.         imx6qdl-sabreauto {

```

```

. . .
895.         pinctrl_usdhc1: usdhc1grp {
896.             fsl,pins = <
897.                 MX6QDL_PAD_SD1_CMD__SD1_CMD 0x17071
898.                 MX6QDL_PAD_SD1_CLK__SD1_CLK 0x10071
899.                 MX6QDL_PAD_SD1_DAT0__SD1_DATA0 0x17071

```

```

. . .
1128.         &usdhc1 { pinctrl-names = "default";
1129.                 pinctrl-0 = <&pinctrl_usdhc1>;
1130.                 cd-gpios = <&gpio1 1 GPIO_ACTIVE_LOW>;
1131.                 no-1-8-v;
1132.                 keep-power-in-suspend;
1133.                 enable-sdio-wakeup;
1134.                 status = "okay";
1135.         };

```

*imx6qdl.dtsi**imx6qdl-sabreauto.dtsi*



Casos de estudio

Bindings

```
542.         gpio1: gpio@0209c000 {
543.             compatible = "fsl,imx6q-gpio", "fsl,imx35-gpio";
544.             reg = <0x0209c000 0x4000>;
545.             interrupts = <0 66 IRQ_TYPE_LEVEL_HIGH>,
546.                         <0 67 IRQ_TYPE_LEVEL_HIGH>;
547.             gpio-controller;
548.             #gpio-cells = <2>;
549.             interrupt-controller;
550.             #interrupt-cells = <2>;
551.         };
```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

→ PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

imx6qdl.dtsi

Casos de estudio

Bindings

```
542.         gpio1: gpio@0209c000 {
543.             compatible = "fsl,imx6q-gpio", "fsl,imx35-gpio";
544.             reg = <0x0209c000 0x4000>;
545.             interrupts = <0 66 IRQ_TYPE_LEVEL_HIGH>,
546.                        <0 67 IRQ_TYPE_LEVEL_HIGH>;
547.             gpio-controller;
548.             #gpio-cells = <2>;
549.             interrupt-controller;
550.             #interrupt-cells = <2>;
551.         };
```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

```
134. static struct platform_device_id mxc_gpio_devtype[] = {
135.     {
136.         .name = "imx1-gpio",
137.         .driver_data = IMX1_GPIO,
138.     }, {
139.         .name = "imx21-gpio",
140.         .driver_data = IMX21_GPIO,
141.     }, {
142.         .name = "imx31-gpio",
143.         .driver_data = IMX31_GPIO,
144.     }, {
145.         .name = "imx35-gpio",
146.         .driver_data = IMX35_GPIO,
147.     }, { /* sentinel */}
148. };
```

imx6qd1.dtsi

.../drivers/gpio/gpio-mxc.c

Casos de estudio

Bindings

```

542.         gpio1: gpio@0209c000 {
543.             compatible = "fsl,imx6q-gpio", "fsl,imx35-gpio";
544.             reg = <0x0209c000 0x4000>;
545.             interrupts = <0 66 IRQ_TYPE_LEVEL_HIGH,
546.                          <0 67 IRQ_TYPE_LEVEL_HIGH>;
547.             gpio-controller;
548.             #gpio-cells = <2>;
549.             interrupt-controller;
550.             #interrupt-cells = <2>;
551.         };

```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

```

134. static struct platform_device_id mxc_gpio_devtype[] = {
135.     {
136.         .name = "imx1-gpio",
137.         .driver_data = IMX1_GPIO,
138.     }, {
139.         .name = "imx21-gpio",
140.         .driver_data = IMX21_GPIO,
141.     }, {
142.         .name = "imx31-gpio",
143.         .driver_data = IMX31_GPIO,
144.     }, {
145.         .name = "imx35-gpio",
146.         .driver_data = IMX35_GPIO,
147.     }, { /* sentinel */
148. };

```

termina siendo utilizado por
module_platform_driver
platform_driver_register

imx6qdl.dtsi

.../drivers/gpio/gpio-mxc.c

.../include/linux/platform_device.h

Casos de estudio

iMX6

```
304     esai: esai@02024000 {
305         #sound-dai-cells = <0>;
306         compatible = "fsl,imx35-esai";
307         reg = <0x02024000 0x4000>;
308         interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309         clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310         <&clks IMX6QDL_CLK_ESAI_MEM>,
311         <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312         <&clks IMX6QDL_CLK_ESAI_IPG>,
313         <&clks IMX6QDL_CLK_SPBA>;
314         clock-names = "core", "mem", "extal", "fsys", "dma";
315         dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316         dma-names = "rx", "tx";
317         status = "disabled";
318     };
```

→ LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

imx6qdl.dtsi

Casos de estudio

iMX6

```

304     esai: esai@02024000 {
305         #sound-dai-cells = <0>;
306         compatible = "fsl,imx35-esai";
307         reg = <0x02024000 0x4000>;
308         interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309         clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310         <&clks IMX6QDL_CLK_ESAI_MEM>,
311         <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312         <&clks IMX6QDL_CLK_ESAI_IPG>,
313         <&clks IMX6QDL_CLK_SPBA>;
314         clock-names = "core", "mem", "extal", "fsys", "dma";
315         dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316         dma-names = "rx", "tx";
317         status = "disabled";
318     };

```

→ LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

./sys/firmware/devicetree/base/soc/aips-bus@02000000/spba-bus@02000000/esai@02024000

./proc/device-tree/sound/cpu-dai

./sys/bus/platform/devices/esai@02024000

./sys/bus/platform/drivers/fsl-esai-dai

./sys/bus/platform/drivers/fsl-esai-dai/esai@02024000

./sys/devices/soc/soc0/soc.0/2000000.aips-bus/2000000.spba-bus/2024000.esai

Casos de estudio

iMX6

```
304         esai: esai@02024000 {
305             #sound-dai-cells = <0>;
306             compatible = "fsl,imx35-esai";
307             reg = <0x02024000 0x4000>;
308             interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309             clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                 <&clks IMX6QDL_CLK_ESAI_MEM>,
311                 <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                 <&clks IMX6QDL_CLK_ESAI_IPG>,
313                 <&clks IMX6QDL_CLK_SPBA>;
314             clock-names = "core", "mem", "extal", "fsys", "dma";
315             dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316             dma-names = "rx", "tx";
317             status = "disabled";
318         };
```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

→ PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

imx6qdl.dtsi

Casos de estudio

iMX6

```

304         esai: esai@02024000 {
305             #sound-dai-cells = <0>;
306             compatible = "fsl,imx35-esai";
307             reg = <0x02024000 0x4000>;
308             interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309             clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                 <&clks IMX6QDL_CLK_ESAI_MEM>,
311                 <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                 <&clks IMX6QDL_CLK_ESAI_IPG>,
313                 <&clks IMX6QDL_CLK_SPBA>;
314             clock-names = "core", "mem", "extal", "fsys", "dma";
315             dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316             dma-names = "rx", "tx";
317             status = "disabled";
318         };

```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

→ PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

```

static const struct of_device_id fsl_esai_dt_ids[] = {
    { .compatible = "fsl,imx35-esai", },
    { .compatible = "fsl,vf610-esai", },
    {}
};
MODULE_DEVICE_TABLE(of, fsl_esai_dt_ids);

```

/sound/soc/fsl/fsl_esai.c

imx6qdl.dtsi

Casos de estudio

iMX6

```

304         esai: esai@02024000 {
305             #sound-dai-cells = <0>;
306             compatible = "fsl,imx35-esai";
307             reg = <0x02024000 0x4000>;
308             interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309             clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                 <&clks IMX6QDL_CLK_ESAI_MEM>,
311                 <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                 <&clks IMX6QDL_CLK_ESAI_IPG>,
313                 <&clks IMX6QDL_CLK_SPBA>;
314             clock-names = "core", "mem", "extal", "fsys", "dma";
315             dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316             dma-names = "rx", "tx";
317             status = "disabled";
318         };

```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

→ PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

```

static const struct of_device_id fsl_esai_dt_ids[] = {
    { .compatible = "fsl,imx35-esai", },
    { .compatible = "fsl,vf610-esai", },
    {}
};

```

```

MODULE_DEVICE_TABLE(of, fsl_esai_dt_ids);

```

```

static struct platform_driver fsl_esai_driver = {
    .probe = fsl_esai_probe,
    .driver = {
        .name = "fsl-esai-dai",
        .pm = &fsl_esai_pm_ops,
        .of_match_table = fsl_esai_dt_ids,
    },
};

```

```

};
module_platform_driver(fsl_esai_driver);

```

/sound/soc/fsl/fsl_esai.c

imx6qdl.dtsi

Casos de estudio

iMX6

```
304         esai: esai@02024000 {
305             #sound-dai-cells = <0>;
306             compatible = "fsl,imx35-esai";
307             reg = <0x02024000 0x4000>;
308             interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309             clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                 <&clks IMX6QDL_CLK_ESAI_MEM>,
311                 <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                 <&clks IMX6QDL_CLK_ESAI_IPG>,
313                 <&clks IMX6QDL_CLK_SPBA>;
314             clock-names = "core", "mem", "extal", "fsys", "dma";
315             dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316             dma-names = "rx", "tx";
317             status = "disabled";
318         };
```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

→ PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

imx6qdl.dtsi

Casos de estudio

iMX6

```

304         esai: esai@02024000 {
305             #sound-dai-cells = <0>;
306             compatible = "fsl,imx35-esai";
307             reg = <0x02024000 0x4000>;
308             interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309             clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                   <&clks IMX6QDL_CLK_ESAI_MEM>,
311                   <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                   <&clks IMX6QDL_CLK_ESAI_IPG>,
313                   <&clks IMX6QDL_CLK_SPBA>;
314             clock-names = "core", "mem", "extal", "fsys", "dma";
315             dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316             dma-names = "rx", "tx";
317             status = "disabled";
318         };

```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

→ PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

En base a lo indicado por esta propiedad `<controlador>_probe`
debe inicializar los registros y la memoria utilizada por el controlador

```
static int fsl_esai_probe(struct platform_device *pdev)
```

}/sound/soc/fsl/fsl_esai.c

Casos de estudio

iMX6

```
304         esai: esai@02024000 {
305             #sound-dai-cells = <0>;
306             compatible = "fsl,imx35-esai";
307             reg = <0x02024000 0x4000>;
308             interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309             clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                 <&clks IMX6QDL_CLK_ESAI_MEM>,
311                 <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                 <&clks IMX6QDL_CLK_ESAI_IPG>,
313                 <&clks IMX6QDL_CLK_SPBA>;
314             clock-names = "core", "mem", "extal", "fsys", "dma";
315             dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316             dma-names = "rx", "tx";
317             status = "disabled";
318         };
```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

→ PROPERTY: IRQ asociada al dispositivo

imx6qdl.dtsi

Casos de estudio

iMX6

```

304             esai: esai@02024000 {
305                 #sound-dai-cells = <0>;
306                 compatible = "fsl,imx35-esai";
307                 reg = <0x02024000 0x4000>;
308                 interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309                 clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                 <&clks IMX6QDL_CLK_ESAI_MEM>,
311                 <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                 <&clks IMX6QDL_CLK_ESAI_IPG>,
313                 <&clks IMX6QDL_CLK_SPBA>;
314                 clock-names = "core", "mem", "extal", "fsys", "dma";
315                 dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316                 dma-names = "rx", "tx";
317                 status = "disabled";
318             };

```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

→ **PROPERTY:** IRQ asociada al dispositivo

0 Indica si es SPI (Shared Peripheral Interrupt) [0=no] (*depende de interpretación del fabricante*)

51 Indica el numero de interrupción. Hay que sumarle 32 si no es SPI o 16 si es SPI para obtener el numero indicado en el manual, 81 en este caso (*explicaciones de esta convención son bienvenidas*)

IRQ_TYPE_LEVEL_HIGH Indica el tipo de detección

imx6qdl.dtsi

Casos de estudio

iMX6

```

304         esai: esai@02024000 {
305             #sound-dai-cells = <0>;
306             compatible = "fsl,imx35-esai";
307             reg = <0x02024000 0x4000>;
308             interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309             clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                  <&clks IMX6QDL_CLK_ESAI_MEM>,
311                  <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                  <&clks IMX6QDL_CLK_ESAI_IPG>,
313                  <&clks IMX6QDL_CLK_SPBA>;
314             clock-names = "core", "mem", "extal", "fsys", "dma";
315             dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316             dma-names = "rx", "tx";
317             status = "disabled";
318         };

```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

PROPERTY: IRQ asociada al dispositivo

→ PROPERTY: Fuentes de reloj asociadas.

imx6qdl.dtsi

Casos de estudio

iMX6

```

304         esai: esai@02024000 {
305             #sound-dai-cells = <0>;
306             compatible = "fsl,imx35-esai";
307             reg = <0x02024000 0x4000>;
308             interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309             clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                  <&clks IMX6QDL_CLK_ESAI_MEM>,
311                  <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                  <&clks IMX6QDL_CLK_ESAI_IPG>,
313                  <&clks IMX6QDL_CLK_SPBA>;
314             clock-names = "core", "mem", "extal", "fsys", "dma";
315             dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316             dma-names = "rx", "tx";
317             status = "disabled";
318         };

```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

PROPERTY: IRQ asociada al dispositivo

PROPERTY: Fuentes de reloj asociadas. Se relaciona a un *phandle* en un fichero *.dts* o a un *.c*

```
static struct clk *clks[IMX6SL_CLK_END];    /arch/arm/mach-imx/clk-imx6sl.c
```

```
struct clk {                                /include/dt-bindings/clock/imx6qdl-clock.h
```

```

struct clk_core*core;
const char *dev_id;
const char *con_id;
unsigned long min_rate;
unsigned long max_rate;
struct hlist_node clks_node;};

```

/drivers/clk/clk.c

imx6qdl.dtsi

Casos de estudio

iMX6

```

304         esai: esai@02024000 {
305             #sound-dai-cells = <0>;
306             compatible = "fsl,imx35-esai";
307             reg = <0x02024000 0x4000>;
308             interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309             clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                  <&clks IMX6QDL_CLK_ESAI_MEM>,
311                  <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                  <&clks IMX6QDL_CLK_ESAI_IPG>,
313                  <&clks IMX6QDL_CLK_SPBA>;
314             clock-names = "core", "mem", "extal", "fsys", "dma";
315             dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316             dma-names = "rx", "tx";
317             status = "disabled";
318         };

```

LABEL: NODE NAME: Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

PROPERTY: IRQ asociada al dispositivo

→ PROPERTY: Fuentes de reloj asociadas. Se relaciona a un *phandle* en un fichero *.dts* o a un *.c*

Nombre amigable con el cual se seleccionara el reloj asociado al dispositivo. El orden debe coincidir con la enumeración (core->IMX6QDL_CLK_ESAI_IPG, mem->IMX6QDL_CLK_ESAI_MEM, etc)

imx6qdl.dtsi

Casos de estudio

iMX6

```

304         esai: esai@02024000 {
305             #sound-dai-cells = <0>;
306             compatible = "fsl,imx35-esai";
307             reg = <0x02024000 0x4000>;
308             interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309             clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                   <&clks IMX6QDL_CLK_ESAI_MEM>,
311                   <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                   <&clks IMX6QDL_CLK_ESAI_IPG>,
313                   <&clks IMX6QDL_CLK_SPBA>;
314             clock-names = "core", "mem", "extal", "fsys", "dma";
315             dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316             dma-names = "rx", "tx";
317             status = "disabled";
318         };

```

LABEL: NODE NAME : Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

PROPERTY: IRQ asociada al dispositivo

PROPERTY: Fuentes de reloj asociadas. Se relaciona a un *phandle* en un fichero *.dts* o a un *.c*

→ PROPERTY: *phandle* al nodo DMA, los canales a utilizar (**¿ideas respecto al 0?**) y el *friendly-name*

imx6qdl.dtsi

Casos de estudio

iMX6

```

304         esai: esai@02024000 {
305             #sound-dai-cells = <0>;
306             compatible = "fsl,imx35-esai";
307             reg = <0x02024000 0x4000>;
308             interrupts = <0 51 IRQ_TYPE_LEVEL_HIGH>;
309             clocks = <&clks IMX6QDL_CLK_ESAI_IPG>,
310                 <&clks IMX6QDL_CLK_ESAI_MEM>,
311                 <&clks IMX6QDL_CLK_ESAI_EXTAL>,
312                 <&clks IMX6QDL_CLK_ESAI_IPG>,
313                 <&clks IMX6QDL_CLK_SPBA>;
314             clock-names = "core", "mem", "extal", "fsys", "dma";
315             dmas = <&sdma 23 21 0>, <&sdma 24 21 0>;
316             dma-names = "rx", "tx";
317             status = "disabled";
318         };

```

LABEL: NODE NAME : Etiqueta que permita asociar un nodo y su correspondiente dirección en el mapa de memoria (I/O)

PROPERTY: Identificador que le permite al SO asociar un dispositivo con el controlador

PROPERTY: Dirección de acceso inicial y rango asociado a los registros de acceso del dispositivo

PROPERTY: IRQ asociada al dispositivo

PROPERTY: Fuentes de reloj asociadas. Se relaciona a un *phandle* en un fichero *.dts* o a un *.c*

→ PROPERTY: *phandle* al nodo DMA, los canales a utilizar (¿ideas respecto al 0?) y el *friendly-name*

imx6qdl.dtsi

Casos de estudio

Bindings

```
17. gpio-keys {
18.     compatible = "gpio-keys1";
19.     pinctrl-names = "default";
20.     pinctrl-0 = <&pinctrl_gpio_keys>;
21.     home {
22.         label = "Home";
23.         gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.         gpio-key;
25.         linux,code = <KEY_HOME>;
26.     };
27.     back {
28.         label = "Back";
29.         gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.         gpio-key;
31.         linux,code = <KEY_BACK>;
32.     };
};
```

imx6qdl.dtsi

Casos de estudio

Bindings

```
17. gpio-keys {
18.     compatible = "gpio-keys1";
19.     pinctrl-names = "default";
20.     pinctrl-0 = <&pinctrl_gpio_keys>;
21.     home {
22.         label = "Home";
23.         gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.         gpio-key;
25.         linux,code = <KEY_HOME>;
26.     };
27.     back {
28.         label = "Back";
29.         gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.         gpio-key;
31.         linux,code = <KEY_BACK>;
32.     };
};
```

➤ BINDING NODE: nombre definido por el usuario

imx6qdl.dtsi

Casos de estudio

Bindings

```
17. gpio-keys {
18.     compatible = "gpio-keys1";
19.     pinctrl-names = "default";
20.     pinctrl-0 = <&pinctrl_gpio_keys>;
21.     home {
22.         label = "Home";
23.         gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.         gpio-key;
25.         linux,code = <KEY_HOME>;
26.     };
27.     back {
28.         label = "Back";
29.         gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.         gpio-key;
31.         linux,code = <KEY_BACK>;
32.     };
};
```

NODE: nombre definido por el usuario

→ PROPERTY (MANDATORIA): modelo de programación para el controlador de teclas input/gpio_keys.c

imx6qdl.dtsi

Casos de estudio

Bindings

```

17.   gpio-keys {
18.       compatible = "gpio-keys1";
19.       pinctrl-names = "default";
20.       pinctrl-0 = <&pinctrl_gpio_keys>;
21.       home {
22.           label = "Home";
23.           gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.           gpio-key;
25.           linux,code = <KEY_HOME>;
26.       };
27.       back {
28.           label = "Back";
29.           gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.           gpio-key;
31.           linux,code = <KEY_BACK>;
32.       };

```

NODE: nombre definido por el usuario

PROPERTY (MANDATORIA): modelo de programación para el controlador de teclas input/gpio_keys.c

equivalente →

```

compatible = "gpio-keys";
#address-cells = <1>;
#size-cells = <0>;

```

imx6qdl.dtsi

Casos de estudio

Bindings

```

17.   gpio-keys {
18.       compatible = "gpio-keys1";
19.       pinctrl-names = "default";
20.       pinctrl-0 = <&pinctrl_gpio_keys>;
21.       home {
22.           label = "Home";
23.           gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.           gpio-key;
25.           linux,code = <KEY_HOME>;
26.       };
27.       back {
28.           label = "Back";
29.           gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.           gpio-key;
31.           linux,code = <KEY_BACK>;
32.       };

```

solo una dirección

NODE: nombre definido por el usuario

PROPERTY (MANDATORIA): modelo de programación para el controlador de teclas input/gpio_keys.c

```

compatible = "gpio-keys";
#address-cells = <1>;
#size-cells = <0>;

```

imx6qdl.dtsi

Casos de estudio

Bindings

```

17.   gpio-keys {
18.       compatible = "gpio-keys1";
19.       pinctrl-names = "default";
20.       pinctrl-0 = <&pinctrl_gpio_keys>;
21.       home {
22.           label = "Home";
23.           gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.           gpio-key;
25.           linux,code = <KEY_HOME>;
26.       };
27.       back {
28.           label = "Back";
29.           gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.           gpio-key;
31.           linux,code = <KEY_BACK>;
32.       };

```

NODE: nombre definido por el usuario

PROPERTY (MANDATORIA): modelo de programación para el controlador de teclas input/gpio_keys.c

```

compatible = "gpio-keys";
#address-cells = <1>;
#size-cells = <0>;

```

La escritura en la dirección actúa sobre la interfaz
(solo una dirección es requerida para el acceso)

imx6qdl.dtsi

Casos de estudio

Bindings

```
17. gpio-keys {
18.     compatible = "gpio-keys1";
19.     pinctrl-names = "default";
20.     pinctrl-0 = <&pinctrl_gpio_keys>;
21.     home {
22.         label = "Home";
23.         gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.         gpio-key;
25.         linux,code = <KEY_HOME>;
26.     };
27.     back {
28.         label = "Back";
29.         gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.         gpio-key;
31.         linux,code = <KEY_BACK>;
32.     };
};
```

NODE: nombre definido por el usuario

PROPERTY: modelo de programación/controlador y dispositivo que representa al nodo

➔ PROPERTY: lista de nombres asociados a un estado. "default=0" (Nombre0=Integer0, Nombre0=Integer1)

imx6qdl.dtsi



Casos de estudio

Bindings

```
17. gpio-keys {
18.     compatible = "gpio-keys1";
19.     pinctrl-names = "default";
20.     pinctrl-0 = <&pinctrl_gpio_keys>;
21.     home {
22.         label = "Home";
23.         gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.         gpio-key;
25.         linux,code = <KEY_HOME>;
26.     };
27.     back {
28.         label = "Back";
29.         gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.         gpio-key;
31.         linux,code = <KEY_BACK>;
32.     };
};
```

NODE: nombre definido por el usuario

PROPERTY: modelo de programación/controlador y dispositivo que representa al nodo

PROPERTY: : lista de nombres asociados a un estado. "default=0" (Nombre0=Integer0, Nombre0=Integer1)

PROPERTY (MANDATORIA): lista de *phandles* que apuntan al modelo de programación/controlador y dispositivo que representa al nodo <pinctrl_gpio_keys>

imx6qdl.dtsi

Casos de estudio

Bindings

```

17. gpio-keys {
18.     compatible = "gpio-keys1";
19.     pinctrl-names = "default";
20.     pinctrl-0 = <&pinctrl_gpio_keys>;
21.     home {
22.         label = "Home";
23.         gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.         gpio-key;
25.         linux,code = <KEY_HOME>;
26.     };
27.     back {
28.         label = "Back";
29.         gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.         gpio-key;
31.         linux,code = <KEY_BACK>;
32.     };

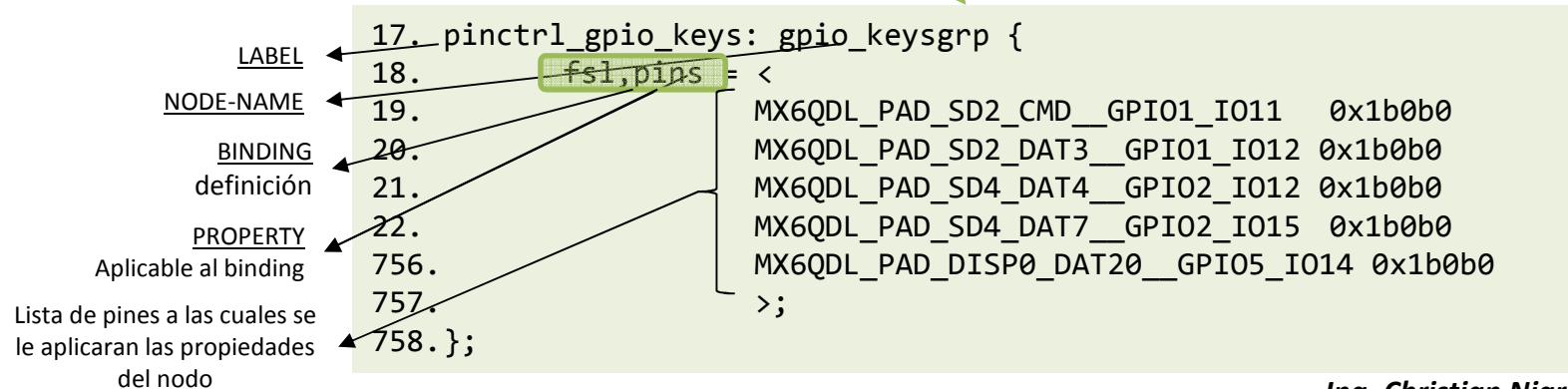
```

NODE: nombre definido por el usuario

PROPERTY: modelo de programación/controlador y dispositivo que representa al nodo

PROPERTY: lista de nombres asociados a un estado. "default=0" (Nombre0=Integer0, Nombre0=Integer1)

PROPERTY (MANDATORIA): lista de *phandles* que apuntan al modelo de programación/controlador y dispositivo que representa al nodo <pinctrl_gpio_keys>



imx6qdl.dtsi

Casos de estudio

Bindings

```

17.   gpio-keys {
18.       compatible = "gpio-keys1";
19.       pinctrl-names = "default";
20.       pinctrl-0 = <&pinctrl_gpio_keys>;
21.       home {
22.           label = "Home";
23.           gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.           gpio-key;
25.           linux,code = <KEY_HOME>;
26.       };
27.       back {
28.           label = "Back";
29.           gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.           gpio-key;
31.           linux,code = <KEY_BACK>;
32.       };

```

NODE: nombre definido por el usuario

PROPERTY: modelo de programación/controlador y dispositivo que representa al nodo

PROPERTY :: lista de nombres asociados a un estado. "default=0" (Nombre0=Integer0, Nombre0=Integer1)

PROPERTY (MANDATORIA): lista de *phandles* que apuntan al modelo de programación/controlador y dispositivo que representa al nodo <pinctrl_gpio_keys>

▶ BINDING NODE: nombre definido por el usuario para el nodo *binding* del dispositivo

imx6qdl.dtsi

Casos de estudio

Bindings

```

17. gpio-keys {
18.     compatible = "gpio-keys1";
19.     pinctrl-names = "default";
20.     pinctrl-0 = <&pinctrl_gpio_keys>;
21.     home {
22.         label = "Home";
23.         gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.         gpio-key;
25.         linux,code = <KEY_HOME>;
26.     };
27.     back {
28.         label = "Back";
29.         gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.         gpio-key;
31.         linux,code = <KEY_BACK>;
32.     };

```

NODE: nombre definido por el usuario

PROPERTY: modelo de programación/controlador y dispositivo que representa al nodo

PROPERTY :: lista de nombres asociados a un estado. "default=0" (Nombre0=Integer0, Nombre0=Integer1)

PROPERTY (MANDATORIA): lista de *handles* que apuntan al modelo de programación/controlador y dispositivo que representa al nodo
<pinctrl_gpio_keys>

BINDING NODE: nombre definido por el usuario para el nodo *binding* del dispositivo

→ PROPERTY: nombre definido por el usuario para el nodo *binding* del dispositivo

imx6qdl.dtsi

Casos de estudio

Bindings

```

17. gpio-keys {
18.     compatible = "gpio-keys1";
19.     pinctrl-names = "default";
20.     pinctrl-0 = <&pinctrl_gpio_keys>;
21.     home {
22.         label = "Home";
23.         gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.         gpio-key;
25.         linux,code = <KEY_HOME>;
26.     };
27.     back {
28.         label = "Back";
29.         gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.         gpio-key;
31.         linux,code = <KEY_BACK>;
32.     };

```

NODE: nombre definido por el usuario

PROPERTY: modelo de programación/controlador y dispositivo que representa al nodo

PROPERTY: : lista de nombres asociados a un estado. "default=0" (Nombre0=Integer0, Nombre0=Integer1)

PROPERTY (MANDATORIA): lista de *handles* que apuntan al modelo de programación/controlador y dispositivo que representa al nodo <pinctrl_gpio_keys>

BINDING NODE: nombre definido por el usuario para el nodo *binding* del dispositivo

PROPERTY: nombre definido por el usuario para el nodo *binding* del dispositivo

→ PROPERTY: indica el grupo (01), el índice dentro del grupo (11) y el estado inicial del GPIO

imx6qdl.dtsi

Casos de estudio

Bindings

```

17. gpio-keys {
18.     compatible = "gpio-keys1";
19.     pinctrl-names = "default";
20.     pinctrl-0 = <&pinctrl_gpio_keys>;
21.     home {
22.         label = "Home";
23.         gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.         gpio-key;
25.         linux,code = <KEY_HOME>;
26.     };
27.     back {
28.         label = "Back";
29.         gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.         gpio-key;
31.         linux,code = <KEY_BACK>;
32.     };

```

NODE: nombre definido por el usuario

PROPERTY: modelo de programación/controlador y dispositivo que representa al nodo

PROPERTY: : lista de nombres asociados a un estado. "default=0" (Nombre0=Integer0, Nombre0=Integer1)

PROPERTY (MANDATORIA): lista de *phandles* que apuntan al modelo de programación/controlador y dispositivo que representa al nodo <pinctrl_gpio_keys>

BINDING NODE: nombre definido por el usuario para el nodo *binding* del dispositivo

PROPERTY: nombre definido por el usuario para el nodo *binding* del dispositivo

PROPERTY: indica el grupo (01), el índice dentro del grupo (11) y el estado inicial del GPIO

→ PROPERTY: indica que el estado es booleano

imx6qdl.dtsi

Casos de estudio

Bindings

```

17. gpio-keys {
18.     compatible = "gpio-keys1";
19.     pinctrl-names = "default";
20.     pinctrl-0 = <&pinctrl_gpio_keys>;
21.     home {
22.         label = "Home";
23.         gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.         gpio-key;
25.         linux,code = <KEY_HOME>;
26.     };
27.     back {
28.         label = "Back";
29.         gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.         gpio-key;
31.         linux,code = <KEY_BACK>;
32.     };

```

NODE: nombre definido por el usuario

PROPERTY: modelo de programación/controlador y dispositivo que representa al nodo

PROPERTY: : lista de nombres asociados a un estado. "default=0" (Nombre0=Integer0, Nombre0=Integer1)

PROPERTY (MANDATORIA): lista de *handles* que apuntan al modelo de programación/controlador y dispositivo que representa al nodo
<pinctrl_gpio_keys>

BINDING NODE: nombre definido por el usuario para el nodo *binding* del dispositivo

PROPERTY: nombre definido por el usuario para el nodo *binding* del dispositivo

PROPERTY: indica el grupo (01), el índice dentro del grupo (11) y el estado inicial del GPIO

PROPERTY: indica que el estado es booleano

→ PROPERTY: indica que el estado es booleano

imx6qdl.dtsi

Casos de estudio

Bindings

```

17. gpio-keys {
18.     compatible = "gpio-keys1";
19.     pinctrl-names = "default";
20.     pinctrl-0 = <&pinctrl_gpio_keys>;
21.     home {
22.         label = "Home";
23.         gpios = <&gpio1 11 GPIO_ACTIVE_LOW>;
24.         gpio-key;
25.         linux,code = <KEY_HOME>;
26.     };
27.     back {
28.         label = "Back";
29.         gpios = <&gpio1 12 GPIO_ACTIVE_LOW>;
30.         gpio-key;
31.         linux,code = <KEY_BACK>;
32.     };

```

NODE: nombre definido por el usuario

PROPERTY: modelo de programación/controlador y dispositivo que representa al nodo

PROPERTY: : lista de nombres asociados a un estado. "default=0" (Nombre0=Integer0, Nombre0=Integer1)

PROPERTY (MANDATORIA): lista de *handles* que apuntan al modelo de programación/controlador y dispositivo que representa al nodo <pinctrl_gpio_keys>

BINDING NODE: nombre definido por el usuario para el nodo *binding* del dispositivo

PROPERTY: nombre definido por el usuario para el nodo *binding* del dispositivo

PROPERTY: indica el grupo (01), el índice dentro del grupo (11) y el estado inicial del GPIO

PROPERTY: indica que el estado es booleano

PROPERTY: indica que el estado es booleano

/drivers/input/keyboard/gpio_keys.c

```

static struct platform_driver gpio_keys_device_driver = {
    .probe= gpio_keys_probe,
    .remove= gpio_keys_remove,
    .driver= {
        .name= "gpio-keys",
        .pm= &gpio_keys_pm_ops,
        .of_match_table = of_match_ptr(gpio_keys_of_match),
    }
};

```



Esto es todo amigos



REFERENCIAS

- Standard for Embedded Power Architecture™ Platform Requirements (ePAPR v1.1 8/04/2011)
- Freescale AN5125 Introduction to Device Trees
- Fernandes J.A. 2013 Flattened Image Trees: A powerful kernel image format
- http://community.cadence.com/cadence_blogs_8/b/sd/archive/2010/04/23/ubuntu-on-arm-is-growing/
- <http://lists.infradead.org/pipermail/linux-armkernel/2011-April/048543.html>
- Linux kernel documentation

Si lo usa
Referencielo

(Un mismo texto puede brindar diferente información según el lector)