

Práctica 1

E/S con espera de respuesta

Puesto de trabajo





- Kit ARM (maletín)
 - Placa Embest S3CEV40
 - LCD + Touchpad +Teclado
 - Interfaz JTAG Olimex
 - Cables de conexión
- Software
 - Entorno basado en Eclipse + OpenOCD
 - Toolchain GNU

Placa Embest S3CEV40

- Basada en un sistema en chip (SoC) de Samsung, el S3C44B0X, que incorpora un procesador ARM7TDMI
- Componentes:
 - Memoria flash de 1M x 16 bits
 - SDRAM de 4 x 1M x 16 bits
 - EEPROM de 4Kbit (IIC)
 - 2 puertos serie (simple y RS-232)
 - 2 botones de interrupción
 - 2 LEDs
 - Interfaz para disco duro IDE
 - Conector JTAG
 - Conector USB

- Ethernet de 10 Mb/s
- LED de 8 segmentos
- Entrada de micrófono
- Salida de altavoces
- CODEC Audio/Voz
- Memoria NAND flash de 16MB

Sistema de memoria

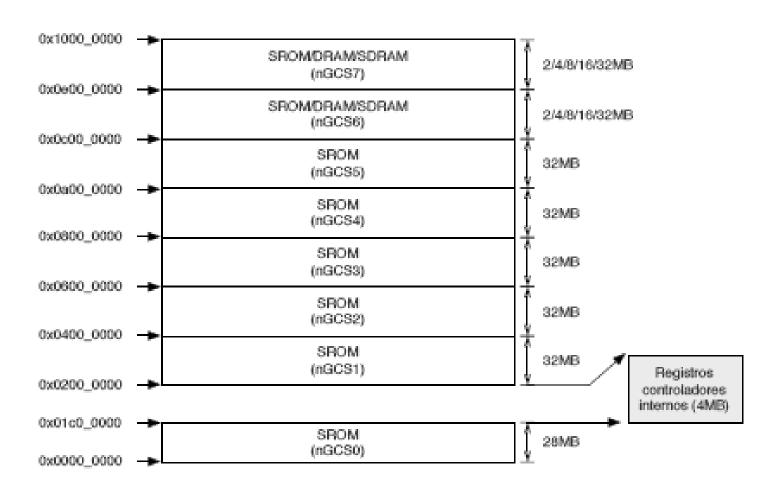
Incluye a los controladores de E/S



- El controlador de memoria
 - Hace de interfaz entre los módulos de memoria externos (ROM o RAM) y el bus del sistema.
 - Reduce el espacio de direcciones efectivo a 256MB y lo divide en 8 bancos. Cada uno puede ser asignado a un chip de memoria externo distinto.
- Comportamiento del controlador de memoria:
 - Cuando el procesador pone una dirección en el bus:
 - Si ésta se encuentra dentro del rango del controlador, se encarga de generar las señales necesarias para realizar el acceso al módulo de memoria que corresponda.
 - Si, por el contrario, la dirección queda fuera de su rango de competencia, el controlador se inhibe, ya que es responsabilidad de algún controlador de E/S atender a dicho acceso.
 - Si el acceso a memoria falla (p.ej. se realiza el acceso a un banco para el que no hay asignado módulo de memoria), el controlador detecta el error y genera la excepción de Abort.

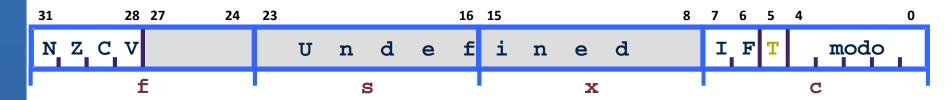
Mapa de memoria

- De todo el espacio de memoria, la parte alta del primer banco, correspondiente al rango 0x01C00000-0x01FFFFFF, está reservada para los puertos de E/S de los controladores integrados dentro del S3C44B0X.
 - Cuando el procesador realiza un acceso dentro este rango el controlador de memoria se inhibe (i.e. no genera señal alguna)



Sistema de E/S del ARM7TDMI

- E/S localizada en memoria
 - Registros especiales (internos) tienen reservado el último tramo del banco-0 [0x01C0 0000-0x01F FFFF]
 - El resto de dispositivos (externos) suelen estar ubicados en el banco-1 [0x0200_0000-0x03FF_FFFF]
 - La selección del dispositivo se realiza externamente (usando el controlador de memoria).
- 1 línea externa de RESET
- 2 líneas de interrupción (IRQ y FIQ)
 - Habilitadas mediante los bit F e I del registro de estado (CPSR)
 - F=1 FIQ deshabilitada
 - I=1 IRQ deshabilitada



Identificación de dispositivos mediante encuesta

Sistema de E/S



- Controlador de pines de E/S (GPIO)
 - 71 pines multifuncionales divididos en 7 grupos
 - 2 grupos de 9 bits de E/S (Puertos E y F)
 - 2 grupos de 8 bits de E/S (Puertos D y G)
 - 1 grupo de 16 bits de E/S (Puerto C)
 - 1 grupo de 11 bits de E/S (Puerto B)
 - 1 grupo de 10 bits de E/S (Puerto A)
 - Existen hasta 3 registros por grupo (control, data y pull-up) que permiten gestionarlos
- Temporizadores
 - 6 temporizadores:
 - 5 con PWM (Pulse Width Modulation) y conexión externa
 - 1 sin PWM y sin conexión externa
 - Pueden trabajar por interrupción o mediante DMA

¿Qué periféricos usaremos?



- LEDs
 - Existen 2 leds
 - Accesibles a través del puerto B de los pines multifunción de E/S (GPIO)
- Pulsadores
 - Existen 2 botones (pulsadores)
 - Accesibles a través del puerto G de los pines multifunción de E/S (GPIO)
- Display 8-segmentos
 - 7 LEDs para formar cualquier dígito hexadecimal
 - 1 LED para el punto decimal
 - Si el LED está a '0' → luz encendida

Sistema de E/S

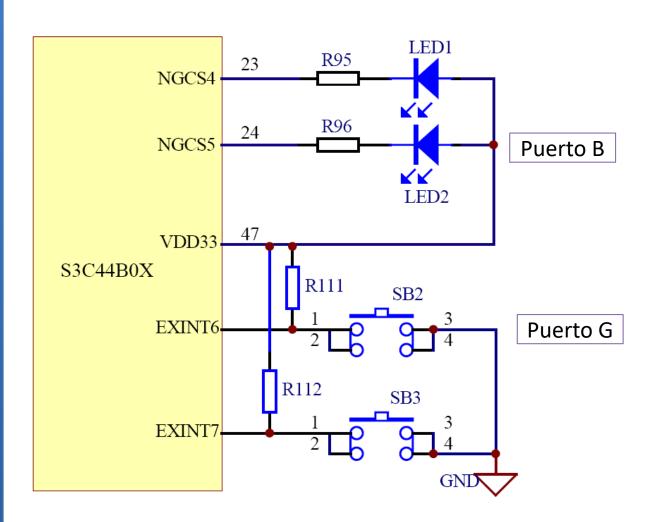


- Dos tipos de dispositivos:
 - Accedidos mediante pines de E/S del S3C44B0X
 - Accedidos mediante direcciones de memoria

Dispositivo	CS	Dirección
USB	CS1	0x0200_0000 - 0x0203_FFFF
Nand Flash	CS2	0x0204_0000 - 0x0207_FFFF
IDE (IOR/W)	CS3	0x0208_0000 - 0x020B_FFFF
IDE (KEY)	CS4	0x020C_0000 - 0x020F_FFFF
IDE (PDIAG)	CS5	0x0210_0000 - 0x0213_FFFF
8-SEG	CS6	0x0214_0000 - 0x0217_FFFF
ETHERNET	CS7	0x0218_0000 - 0x021B_FFFF
LCD	CS8	0x021C_0000 - 0x021F_FFFF
Teclado	nGCS3	0x0600_0000 - 0X07FF_FFFF

Pulsadores y leds









- Puerto B (11 pines)
 - Registro de control de 11 bits: PCONB
 - Registro de datos de 11 bits: PDATB

Registro	Dirección	R/W	Descripción	Valor por defecto
PCONB	0x01D20008	R/W	Configuración pines	0x7ff
PDATB	0x01D2000C	R/W	Datos	Undef

- Con la configuración por defecto
 - PB10 = nGCS5
 - PB9 = nGCS4

Uso de LEDs



- Funcionamiento:
 - 1. Configurar pines 9 y 10 como salida (una vez)
 - Escribir un '0' en los bits 9 y 10 de PCONB
 - Simplificación: escribir todo '0' en PCONB
 - 2. Escribir en PDATB para encender/apagar. Ejemplo:
 - Si bit 9 de PDATB =0 → LED 1 encendido
 - Si bit 10 de PDATB=1 → LED 2 apagado

Conexión de los pulsadores



Puerto G (8 pines)

Registro	Dirección	R/W	Descripción	Valor por defecto
PCONG	0x01D20040	R/W	Configuración pines	0x0
PDATG	0x01D20044	R/W	Datos	Undef
PUPG	0x01D20048	R/W	Deshabilitar pull-up	0x0

Para que funcionen es necesario PUPG=0x0

Uso de pulsadores

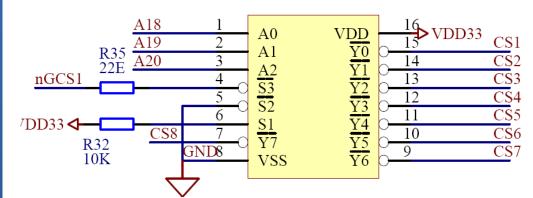
- Conectados a los pines 6 y 7 del puerto G
 - Registro de control de 16 bits (PCONG) en dirección 0x01D20040
 - 2º registro de control de 8 bits PUPG en dirección 0x01D20048
 - Registro de datos de 8 bits (PDATG) en dirección 0x01D20044
- Operación:
 - 1. Configurar pines 6 y 7 como entrada (una vez)
 - Escribir '0000' en los bits 15:12 de PCONG
 - Escribir todo '0' en PUPG
 - 2. Leer de PDATG para comprobar si pulsado. Ejemplo:
 - Si bit $6 = 0 \rightarrow botón 1 pulsado$
 - Si bit $7 = 1 \rightarrow botón 2 NO pulsado$

Acceso a dispositivos mediante direcciones de memoria

Selección de chips

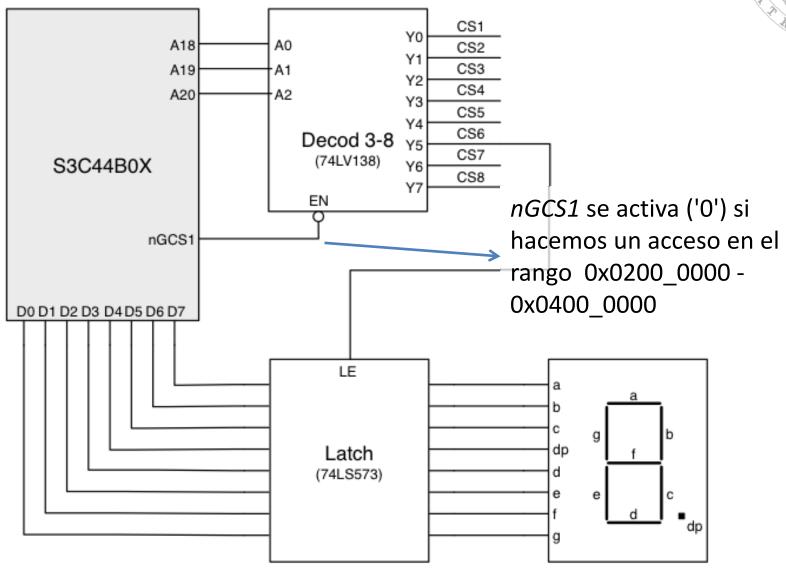
A20	A19	A18	CS	Modulo
0	0	0	CS1	USB
0	0	1	CS2	Nand Flash
0	1	0	CS3	IDE
0	1	1	CS4	IDE
1	0	0	CS5	IDE
1	0	1	CS6	8-SEG
1	1	0	CS7	ETHERNET
1	1	1	CS8	LCD

U7 74LV138



Uso display 8-segmentos

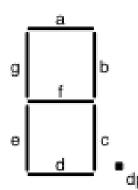




Uso display 8-segmentos

ALIVA

- No existe registro de control (siempre configurados como salida)
- Registro de datos (8 bits) en dirección 0x2140000
 - Cada segmento tiene asociado 1 bit del registro
 - led a -> bit de mayor peso. Orden a b c dp d e f g
 - Si bit₀=0 → el segmento g se enciende



- ¿Qué ocurre si escribimos 0x18?
- ¿Qué hay que escribir para conseguir una C?