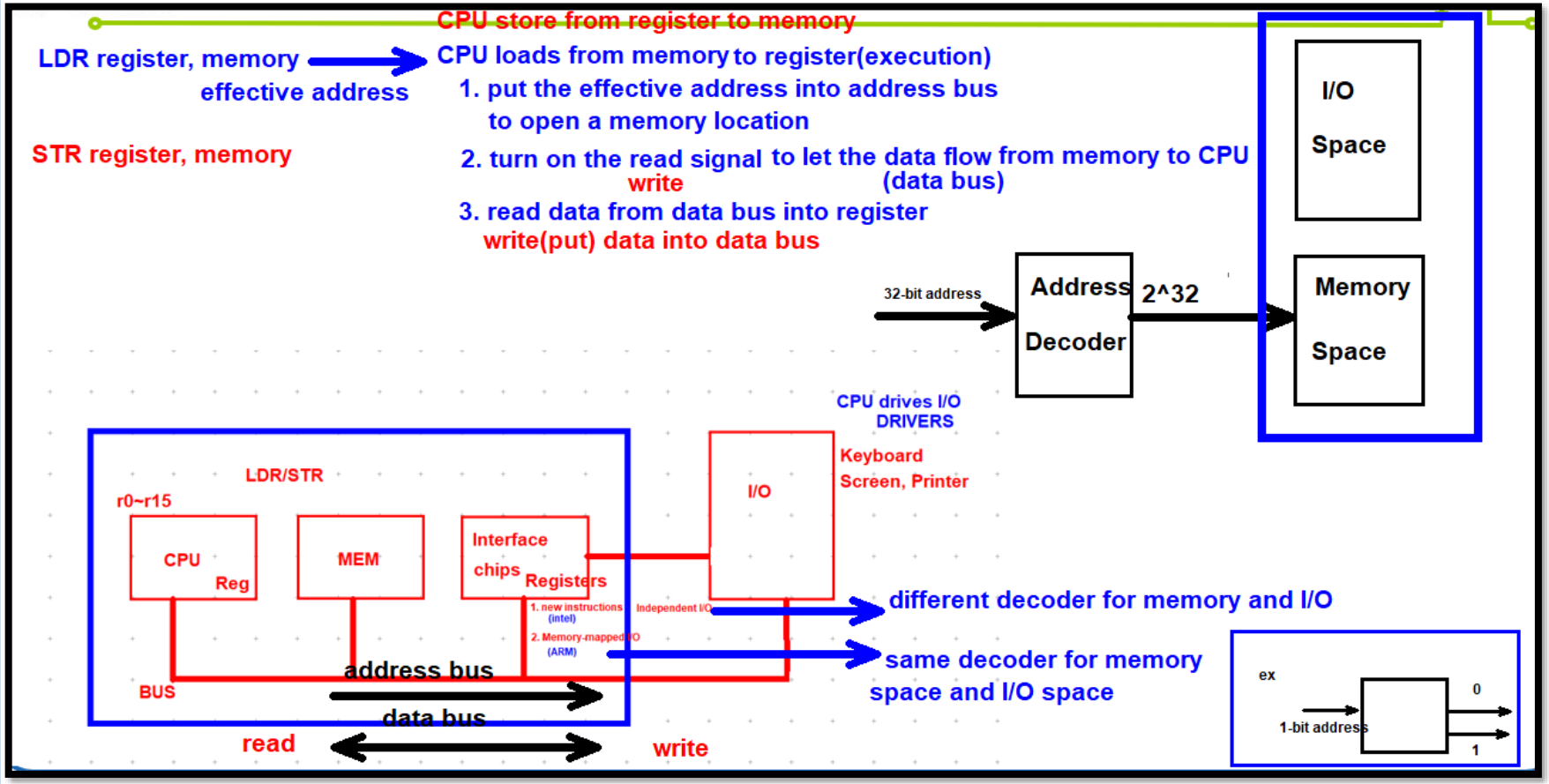
第02次實習課

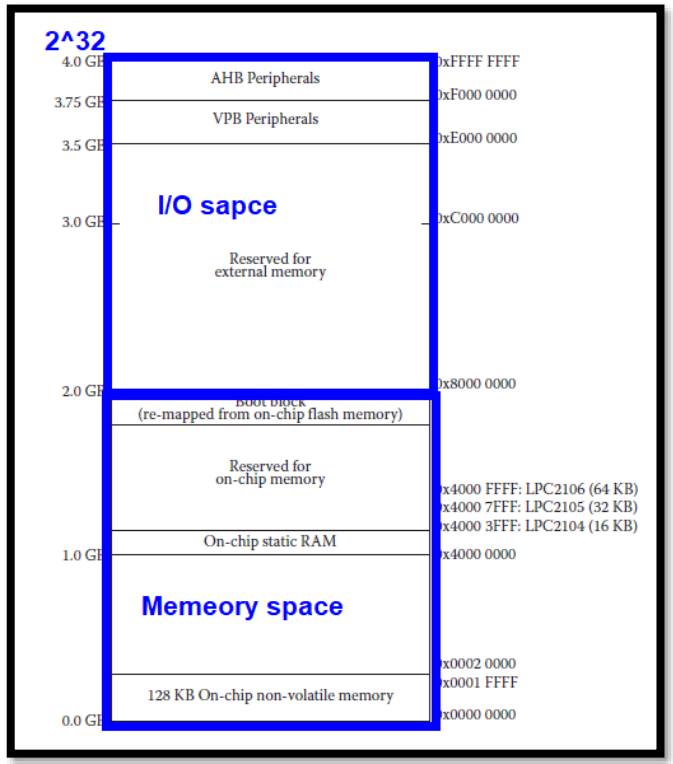
學生：林培瑋

2024 Advanced Mixed-Operation System (AMOS) Lab.

Tamkang University
Department of Electrical and Computer Engineering
No.151, Yingzhuan Rd., Tamsui Dist., New Taipei City 25137, Taiwan (R.O.C.)

簡介


CPU store from register to memory
LDR register, memory
effective address
CPU loads from memory to register(execution)
1. put the effective address into address bus
to open a memory location
STR register, memory
2. turn on the read signal to let the data flow from memory to CPU
write
(data bus)
3. read data from data bus into register
write(put) data into data bus
I/O
Space
Memory
Space
32-bit address
Address
Decoder
2^32
CPU drives I/O
DRIVERS
Keyboard
Screen, Printer
LDR/STR
r0~r15
CPU
Reg
MEM
Interface
chips
Registers
I/O
1. new instructions
(intel)
Independent I/O
2. Memory-mapped I/O
(ARM)
different decoder for memory and I/O
same decoder for memory
space and I/O space
BUS
address bus
data bus
read
write
ex
1-bit address
0
1


2^32
4.0 GB
3.75 GB
3.5 GB
3.0 GB
2.0 GB
1.0 GB
0.0 GB
AHB Peripherals
VPB Peripherals
I/O sapce
Reserved for
external memory
(re-mapped from on-chip flash memory)
Reserved for
on-chip memory
On-chip static RAM
Memeory space
128 KB On-chip non-volatile memory
0xFFFF FFFF
0xF000 0000
0xE000 0000
0xC000 0000
0x8000 0000
0x4000 FFFF: LPC2106 (64 KB)
0x4000 7FFF: LPC2105 (32 KB)
0x4000 3FFF: LPC2104 (16 KB)
0x4000 0000
0x0002 0000
0x0001 FFFF
0x0000 0000

P345 FIGURE 16.3

8-bit registers

UART 0			高							低		初始值
0xE000C000	U0RBR (DLAB=0)	U0 Receiver buffer register	8-bit data								RO	un–defined
	U0THR (DLAB=0)	U0 Transmit holding register	8-bit data								WO	NA
	U0DLL (DLAB=1)	U0 Divisor latch LSB	8-bit data								R/W	0x01
0xE000C004	U0IER (DLAB=0)	U0 Interrupt enable register	0	0	0	0	0	En. Rx Line Status Int.	Enable THRE Int.	En. Rx Data Av.Int.	R/W	0
	U0DLM (DLAB=1)	U0 Divisor latch LSB	8 bit data								R/W	0
0xE000C008	U0IIR	U0 Interrupt ID register	FIFOs Enabled		0	0	IIR3	IIR2	IIR1	IIR0	RO	0x01
	U0FCR	U0 FIFO control register	Rx Trigger		-	-	-	U0 Tx FIFO Reset	U0 Rx FIFO Reset	U0 FIFO Enable	WO	0
0xE000C00C	U0LCR	U0 Line control register	DLAB	Set break	Stick parity	Even parity select	Parity enable	Nm. of stop bits	Word length select		R/W	0
0xE000C014	U0LSR	U0 Line status register	Rx FIFO Error	TEMT	THRE	BI	FE	PE	OE	DR	RO	0x60
0xE000C01C	U0LSR	U0 Scratch pad register	8-bit data								R/W	0

FIGURE 16.3 Memory map of UART0 on the LPC2104. (From LPC2106/2105/2104 User Manual NXP Semiconductors, September 2003. With permission.)

P343 TABLE 16.1

Receiver ← Transmitter

Buffer (Line)

CPU initialize UART(initialization) ROM BIOS(Basic I/O services)

operation

TABLE 16.1 Line formet

UART Configuration Bits in the Control Register

Line Contorl Register

U0LCR	Function	Description	Reset Value
1:0	Word Length Select	00:5-bit character length	0
		01:6-bit character length	
		10:7-bit character length	
		11:8-bit character length	
2	Stop Bit Select	0:1 stop bit	0
		1:2 stop bits (1.5 if U0LCR[1:0] = 00)	
3	Parity Enable	0: Disable parity generation and checking	0
		1: Enable parity generation and checking	
5:4	Parity select	00: Odd parity	0
		01: Even parity	
		10: Forced "l" stick parity	
		11: Forced "0" stick parity	
6	Break Control	0: Disable break transmission	0
		1: Enable break transmission. Output pin UART0 TxD is forced to logic 0 when U0LCR6 is actively high	
7	Divisor Latch Access Bit	0: Disable access to divisor latches	0
		1: Enable access to divisor latches	

P346 程式

The next step to configuring the UART is to set the number of data bits, the parity, and the number of stop bits. Again, the starting address of the UART0 configuration register, 0xE000C000, is loaded into a general register to be used as a base address. The LCR and LSR registers can be accessed using a pre-indexed addressing scheme, where the offsets are equated to known values at the beginning of the final routine. Here, LCR0 would be equated to 0xC, and for our write routine, LSR0 would be equated to 0x14. Since these are 8-bit registers, they must be accessed using STRB and LDRB instructions. The rest of the configuration code is below.

題目：initialize UART to have 8-bit character length , no-parity, 1-stop-bit formet of line.

組合語言：

Write 0x83 into a byte at memory address 0xE000C00C

```
LDR r5, =0xE000C00C
MOV r6, #0x83
STRB r6, [r5]
```

```
LDR     r5, =U0START
MOV     r6, #0x83          ; set 8 bits, no parity, 1 stop bit
STRB    r6, [r5, #LCR0]    ; write control byte to LCR
MOV     r6, #0x61          ; 9600 baud @15 MHz VPB clock
STRB    r6, [r5]           ; store control byte
```

Q&A

Thanks for your attention !!