Memory Controller

Speaker: Tzu-Wei Tseng

Adopted from National Taiwan University
SoC Design Laboratory

Goal of This Lab

- Familiarize with ARM memory interface
- Know ARM Integrator memory map well
- How to access memory

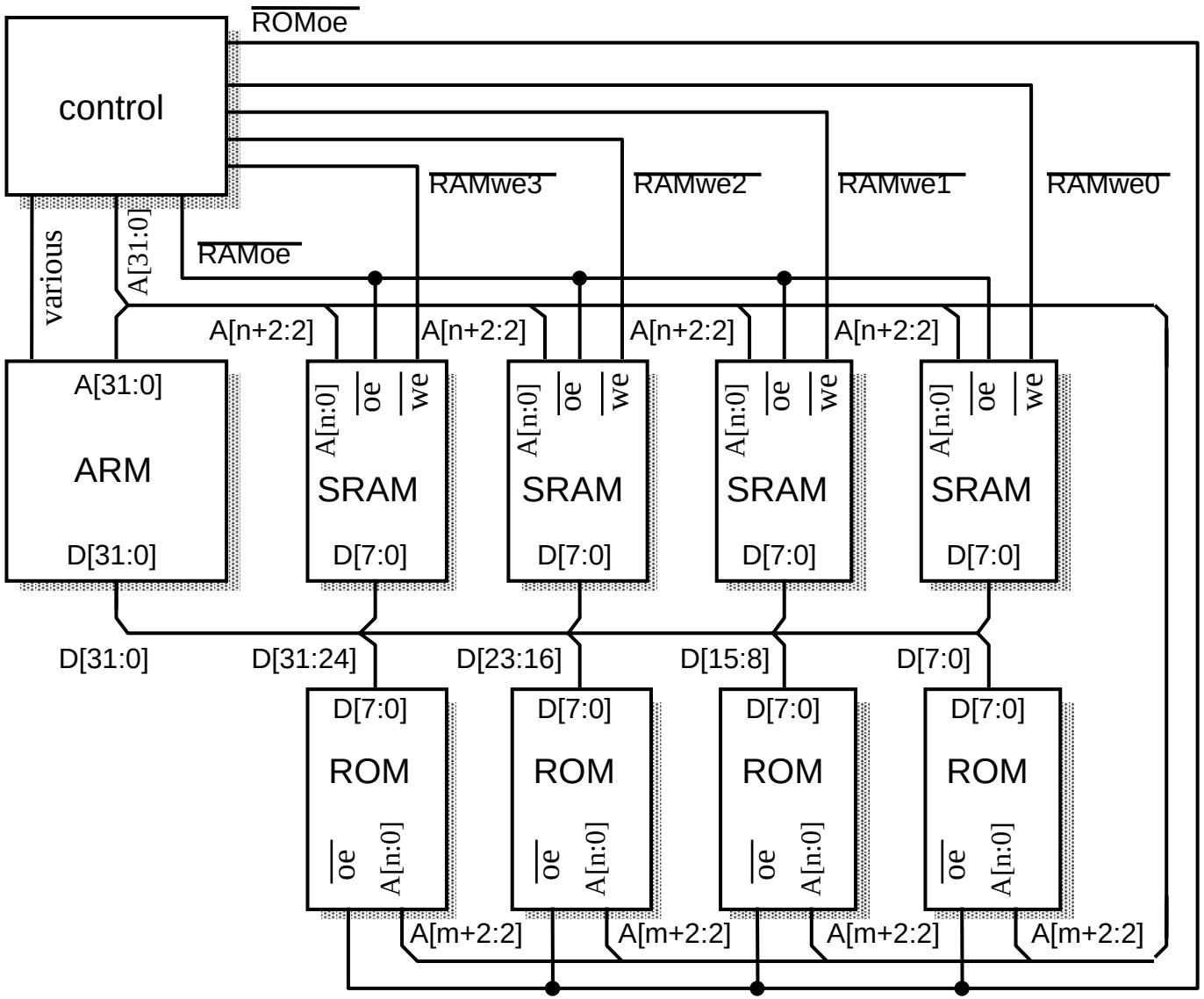
Outline

- ***The ARM Memory Interface [1]***
- ARM Integrator System Memory Map [2] [3]
- Lab – Memory Control

Simple Memory Interface

- The simplest form of memory interface is suitable for operation with **ROM** and static RAM (**SRAM**).
- 8-bit memory types, so **four** parts of each type are required to form a 32-bit memory.
- Since the bottom two address lines, *A[1:0]*, are used for byte selection, they are used by the control logic and not connected to the memory.
- Although the ARM performs reads of both bytes and words, the memory system can ignore the difference and always simply supply a word quantity.

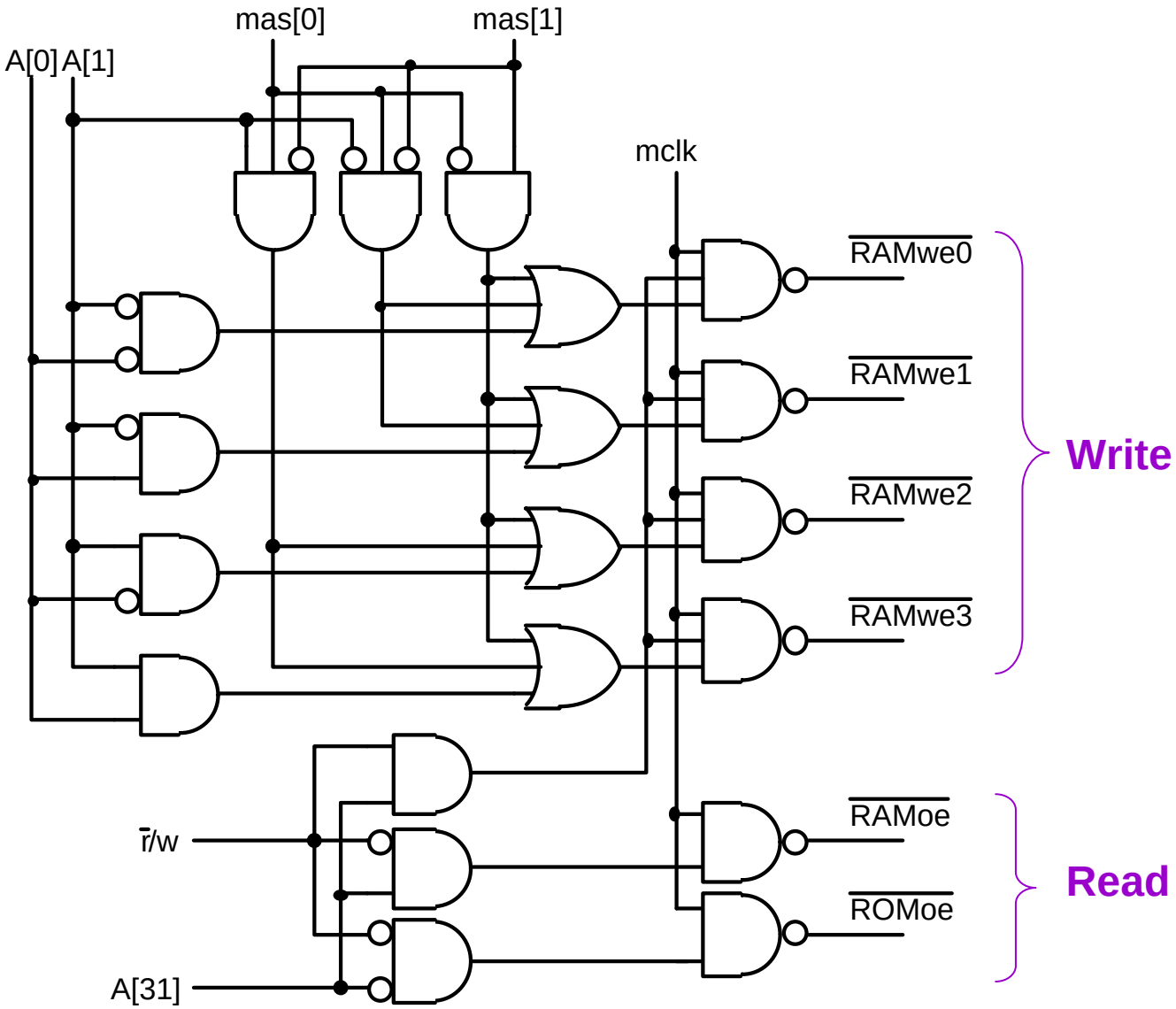
Basic ARM Memory System



Control Logic

- It decides when to activate the ARM and when to activate the ROM
  - A[31] = 0 => access ROM
  - A[30] = 1 => access RAM
- It controls the byte write enables during a write operation
  - Word write / half-word write / byte write
- It ensures that the data is ready before the processor continues
  - The simplest solution is to run *mclk* slowly enough to ensure that all the memory devices can be accessed within a single clock cycle

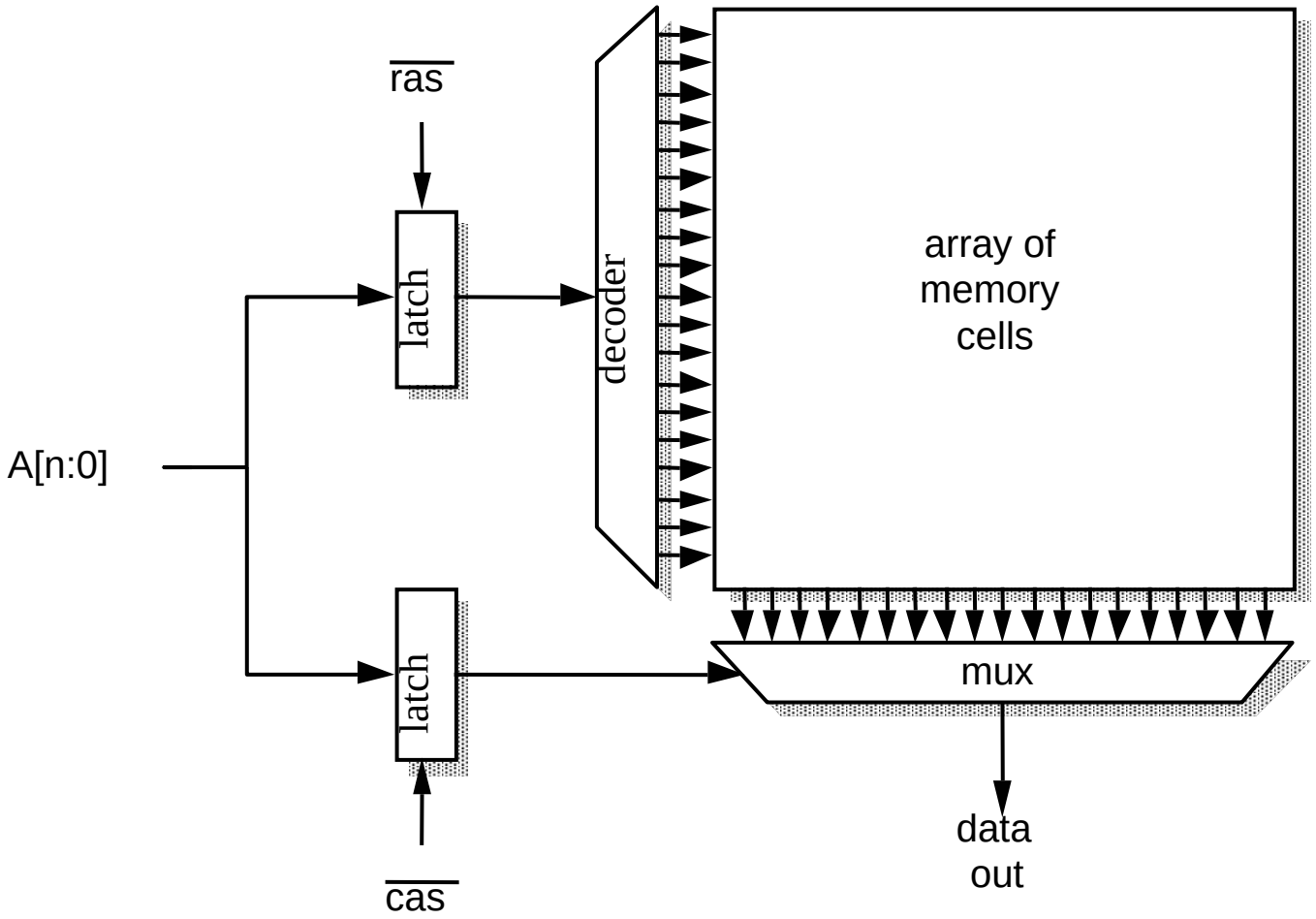
Simple ARM Memory System Control Logic



Transfer Widths

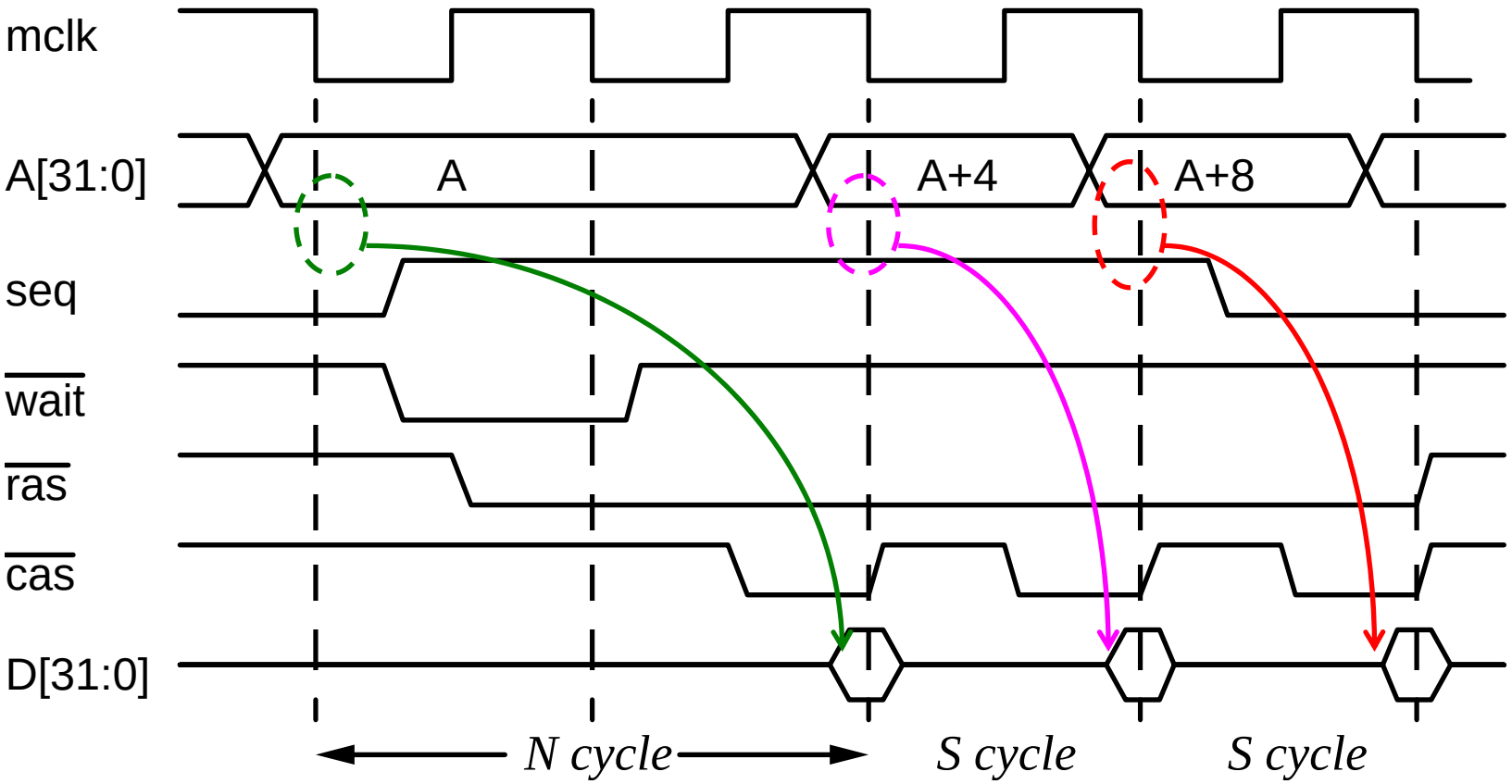
A[31]	$\overline{r}$/w	mas[1]	mas[0]	A[1]	A[0]	Output signal	
0	0	X	X	X	X	$\overline{ROMoe}$	Read
0	1	X	X	X	X	None	
1	0	X	X	X	X	$\overline{RAMoe}$	
1	1	0	0	X	X	None	Write
1	1	1	1	X	X		
1	1	1	0	X	X	Word ($\overline{RAMwe0}$, $\overline{RAMwe1}$, $\overline{RAMwe2}$, $\overline{RAMwe3}$)	
1	1	0	1	0	X	Half word ($\overline{RAMwe0}$, $\overline{RAMwe1}$)	
1	1	0	1	1	X	Half word ($\overline{RAMwe2}$, $\overline{RAMwe3}$)	
1	1	X	X	0	0	Byte ($\overline{RAMwe0}$)	
1	1	X	X	0	1	Byte ($\overline{RAMwe1}$)	
1	1	X	X	1	0	Byte ($\overline{RAMwe2}$)	
1	1	X	X	1	1	Byte ($\overline{RAMwe3}$)	

DRAM memory organization



ras: row address strobe cas: column address strobe

DRAM Timing Illustration



Outline

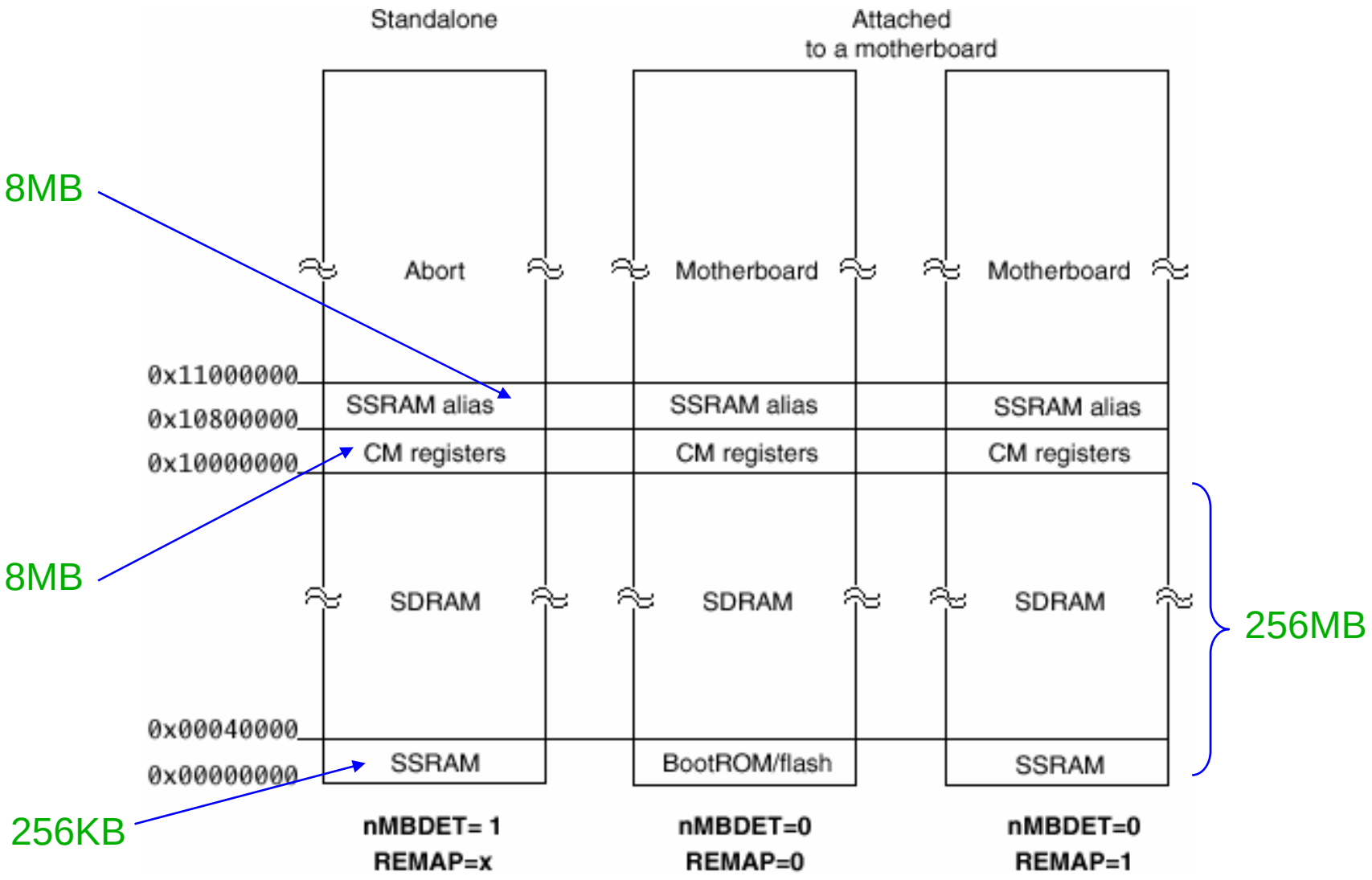
- The ARM Memory Interface [1]
- ***ARM Integrator System Memory Map [2][3]***
- Lab – Memory Control

Core Module Memory Map

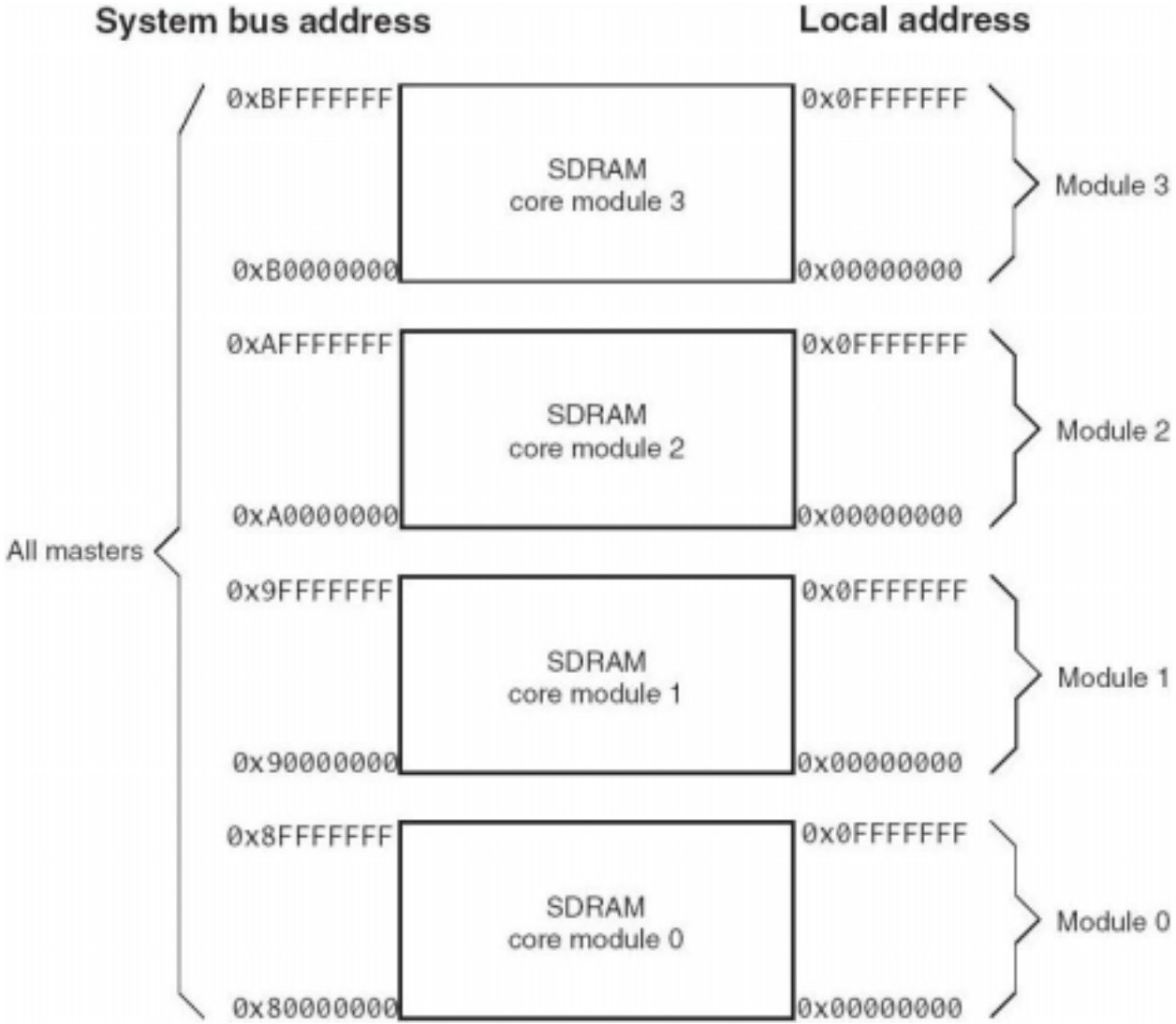
nMBDET	REMAP	Address range	Region size	Description
0	0	0x00000000 to 0x0003FFFF	256KB	Boot ROM (on motherboard)
0	1	0x00000000 to 0x0003FFFF	256KB	SSRAM
1	X	0x00000000 to 0x0003FFFF	256KB	SSRAM
X	X	0x00040000 to 0x0FFFFFFF	256MB	Local SDRAM
X	X	0x10000000 to 0x107FFFFF	8MB	Core Module registers
X	X	0x10800000 to 0x10FFFFFF	8MB	SSRAM alias
0	X	0x11000000 to 0xFFFFFFFF	272MB to 4GB	System bus address space
1	X	0x11000000 to 0xFFFFFFFF	272MB to 4GB	Abort

- The **nMBDET** signal is permanently grounded by the motherboard so that it is pulled LOW on the core module when it is fitted.
- The **REMAP** bit only has effect if the core module is attached to a motherboard (**nMBDET** = 0).

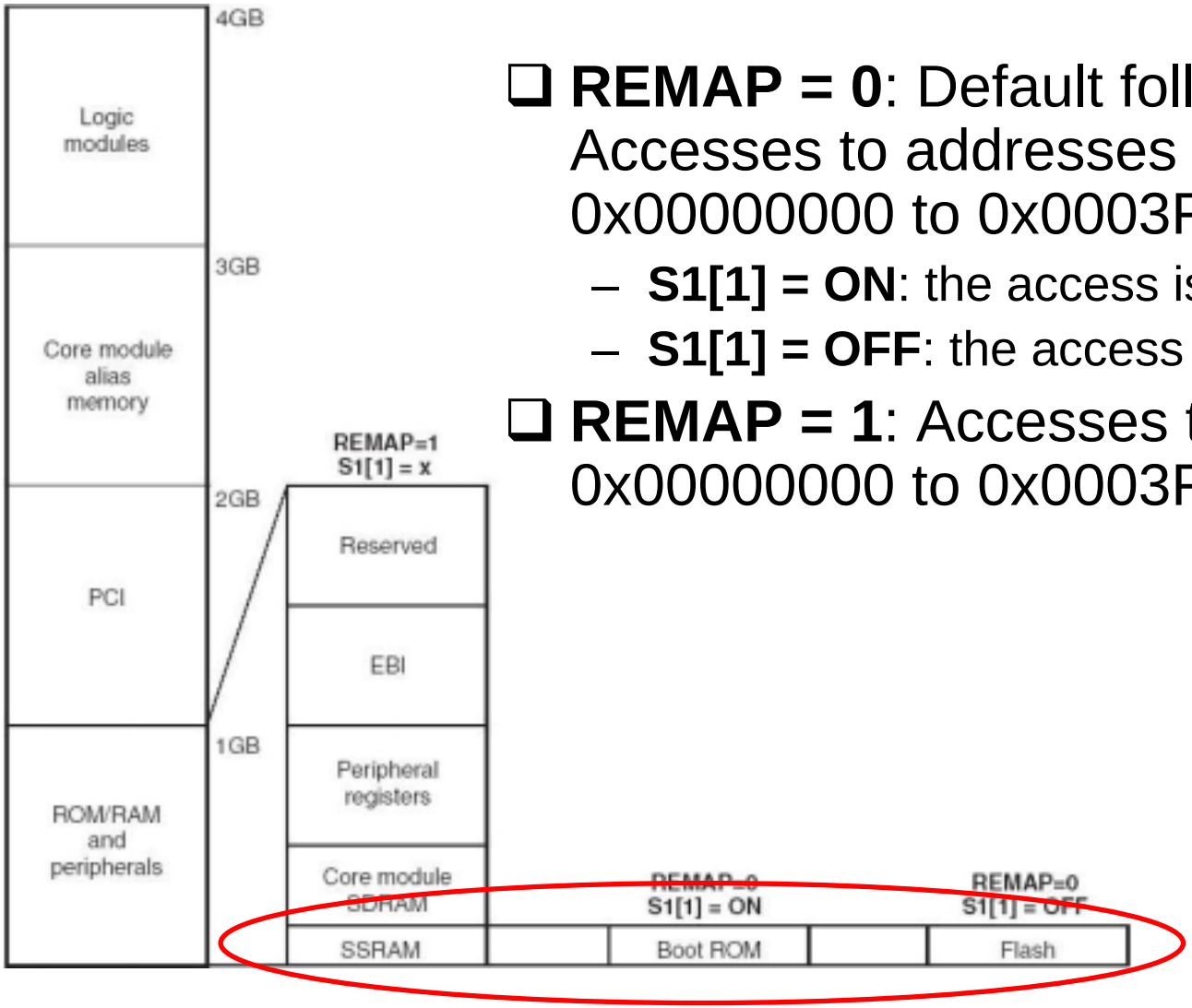
Core Module Memory Map (cont.)



Core Module Alias Address

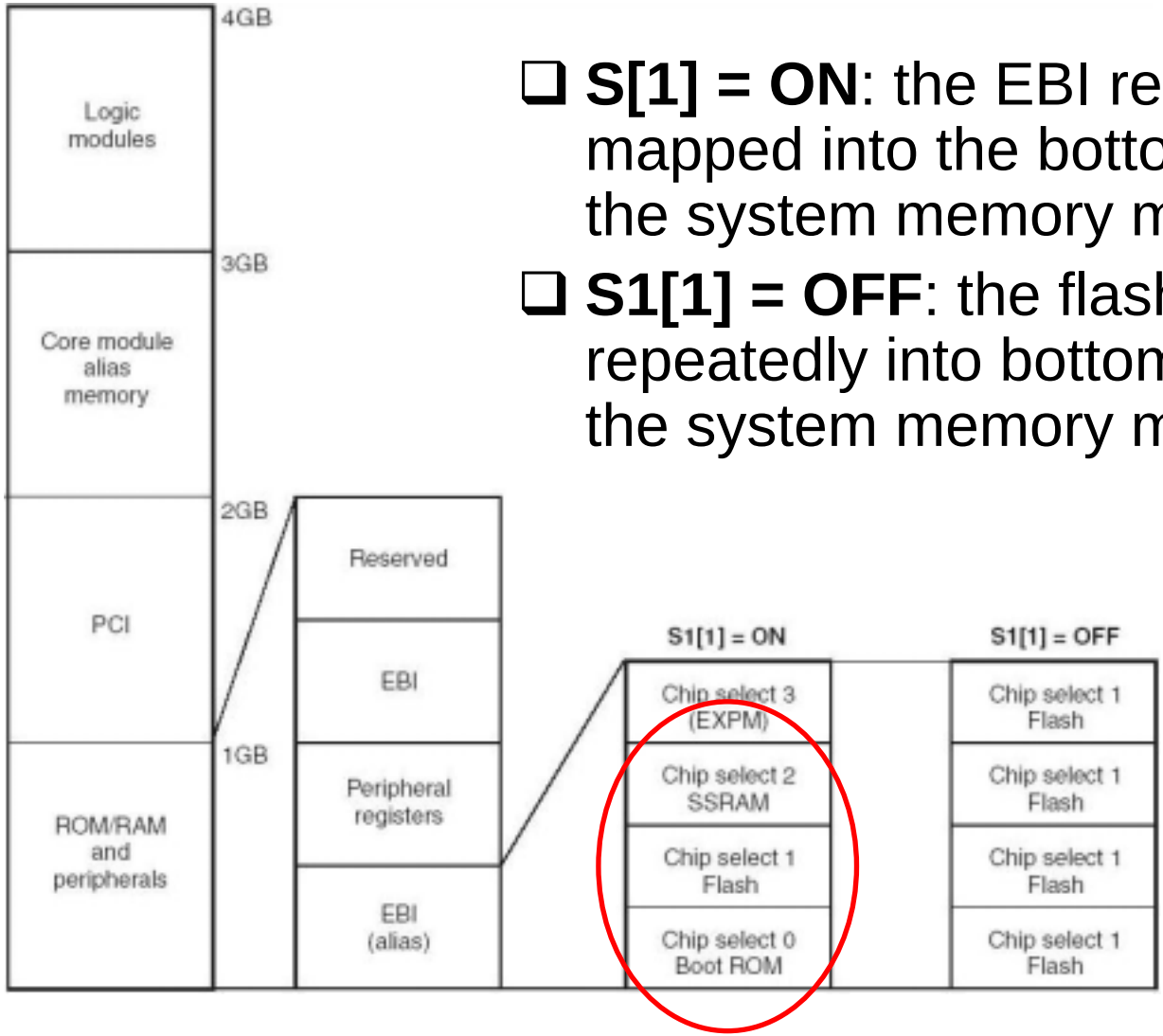


Memory Map for Core Module



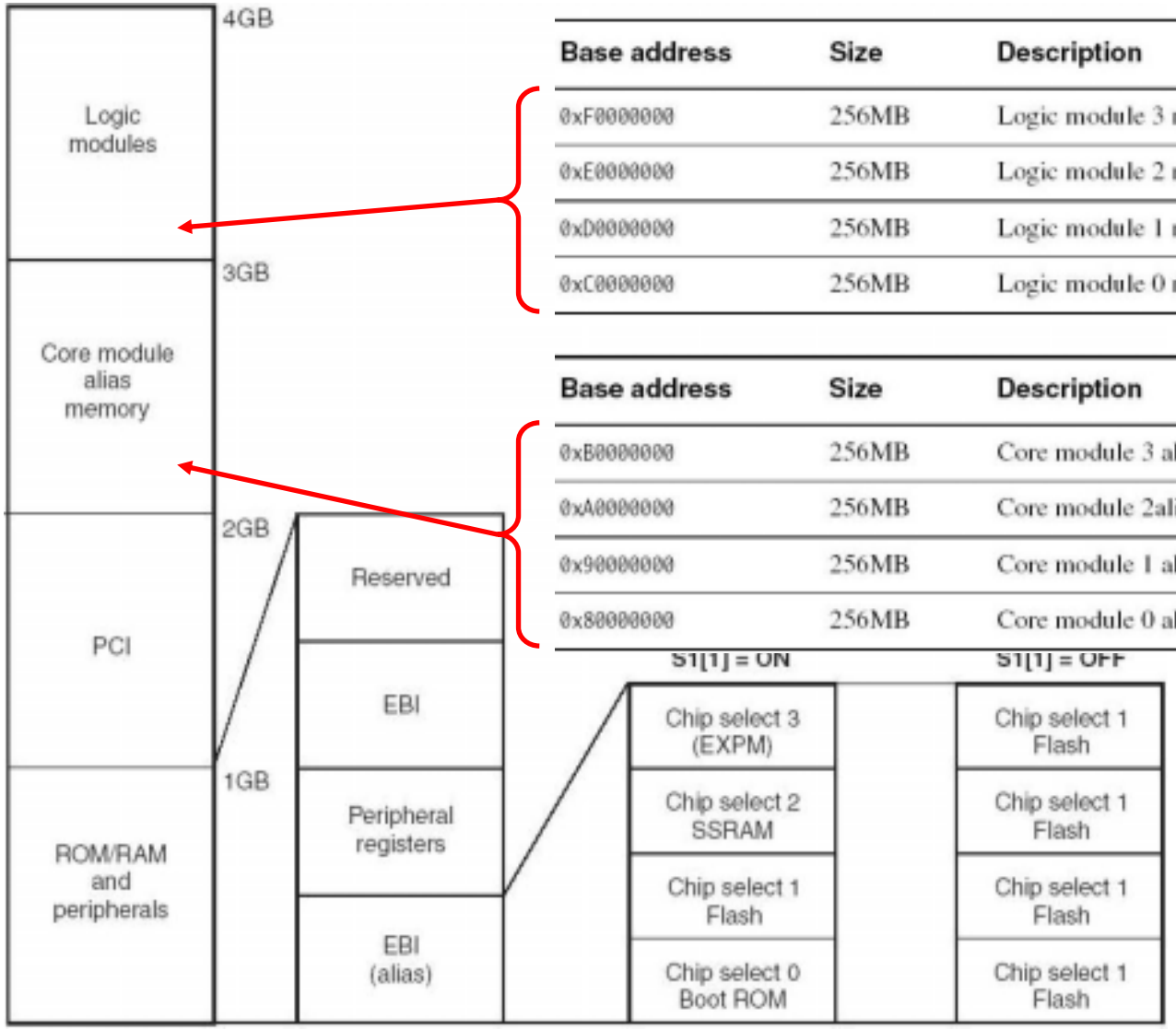
- **REMAP = 0**: Default following reset. Accesses to addresses 0x00000000 to 0x0003FFFF
  - **S1[1] = ON**: the access is to boot ROM
  - **S1[1] = OFF**: the access is to flash
- **REMAP = 1**: Accesses to address 0x00000000 to 0x0003FFFF

Memory Map for Logic Modules



- **S[1] = ON**: the EBI resources are mapped into the bottom 256MB of the system memory map.
- **S1[1] = OFF**: the flash is mapped repeatedly into bottom 256MB of the system memory map.

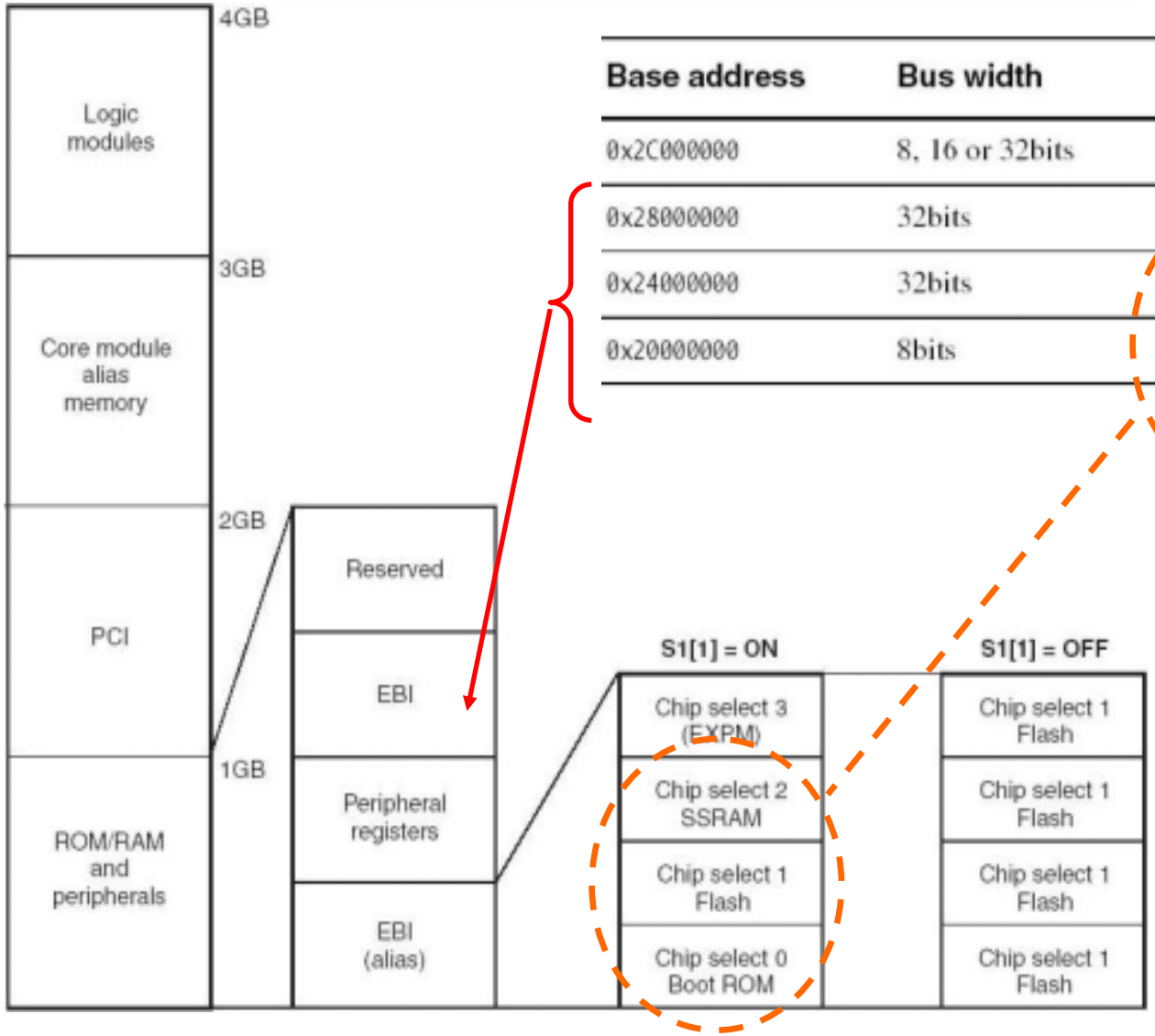
System Memory Map (1/3)



Base address	Size	Description	ID[3:0]
0xF0000000	256MB	Logic module 3 memory	1101
0xE0000000	256MB	Logic module 2 memory	1011
0xD0000000	256MB	Logic module 1 memory	0111
0xC0000000	256MB	Logic module 0 memory	1110

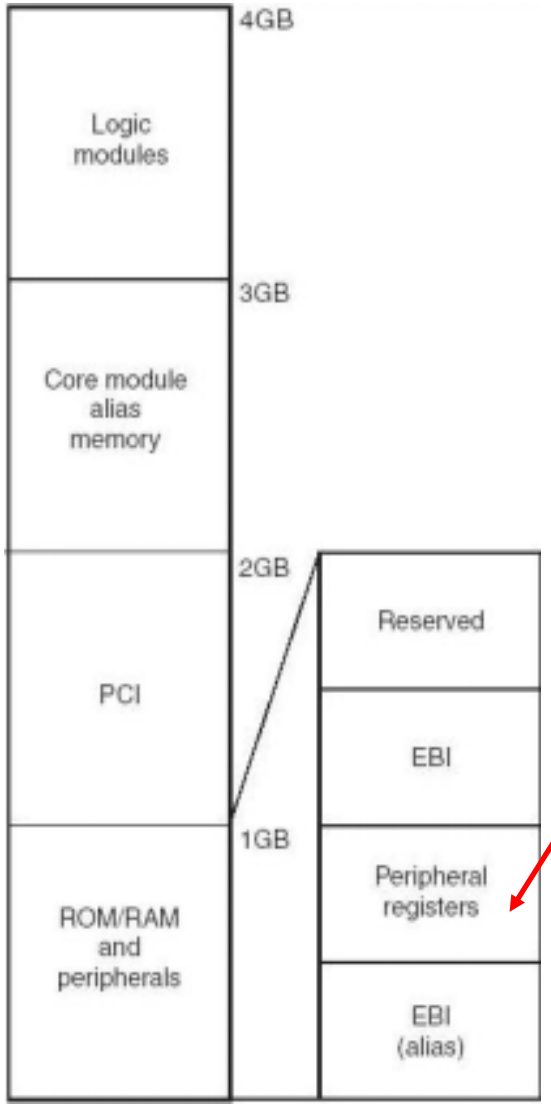
Base address	Size	Description	ID[3:0]
0xB0000000	256MB	Core module 3 alias memory	1101
0xA0000000	256MB	Core module 2alias memory	1011
0x90000000	256MB	Core module 1 alias memory	0111
0x80000000	256MB	Core module 0 alias memory	1110

System Memory Map (2/3)



Base address	Bus width	Description
0x2C000000	8, 16 or 32bits	Chip select 3 (for expansion)
0x28000000	32bits	SSRAM
0x24000000	32bits	Flash
0x20000000	8bits	Boot ROM

System Memory Map (3/3)



Base address	Size	Description
0x1F000000	16MB	Spare
0x1E000000	16MB	Spare
0x1D000000	16MB	Spare
0x1C000000	16MB	Spare
0x1B000000	16MB	GPIO
0x1A000000	16MB	LED display and boot switch
0x19000000	16MB	Mouse
0x18000000	16MB	Keyboard
0x17000000	16MB	UART1
0x16000000	16MB	UART0
0x15000000	16MB	RTC
0x14000000	16MB	Interrupt controller
0x13000000	16MB	Counter/timers
0x12000000	16MB	EBI configuration registers
0x11000000	16MB	System controller registers
0x10000000	16MB	Core module registers (for core modules) Spare (for logic modules)

Outline

- The ARM Memory Interface
- ARM Integrator System Memory Map
- ***Lab – Memory Control***

Lab 7: Memory Controller

- Goal
  - Realize the principle of memory map and internal and external memory
- Principles
  - System memory map
  - Core Module Control Register
  - Core Module Memory Map
- Guidance
  - We use a simple program to lead student understanding the memory.
- Requirements and Exercises
  - Compare the performance between using SSRAM and SDRAM
- Discussion
  - Discuss the following items about Flash, RAM, and ROM.
    - Speed
    - Capacity
    - Internal /External

References

[1] http://access.ee.ntu.edu.tw/course/SOC_LAB/index.html

[1] **ARM System-on-Chip Architecture** by S.Furber, Addison Wesley Longman: ISBN 0-201-67519-6.

[2] DUI0098B_AP_UG.pdf.

[3] DUI0126B_CM7TDMI_UG.pdf