

ARM内核结构简介

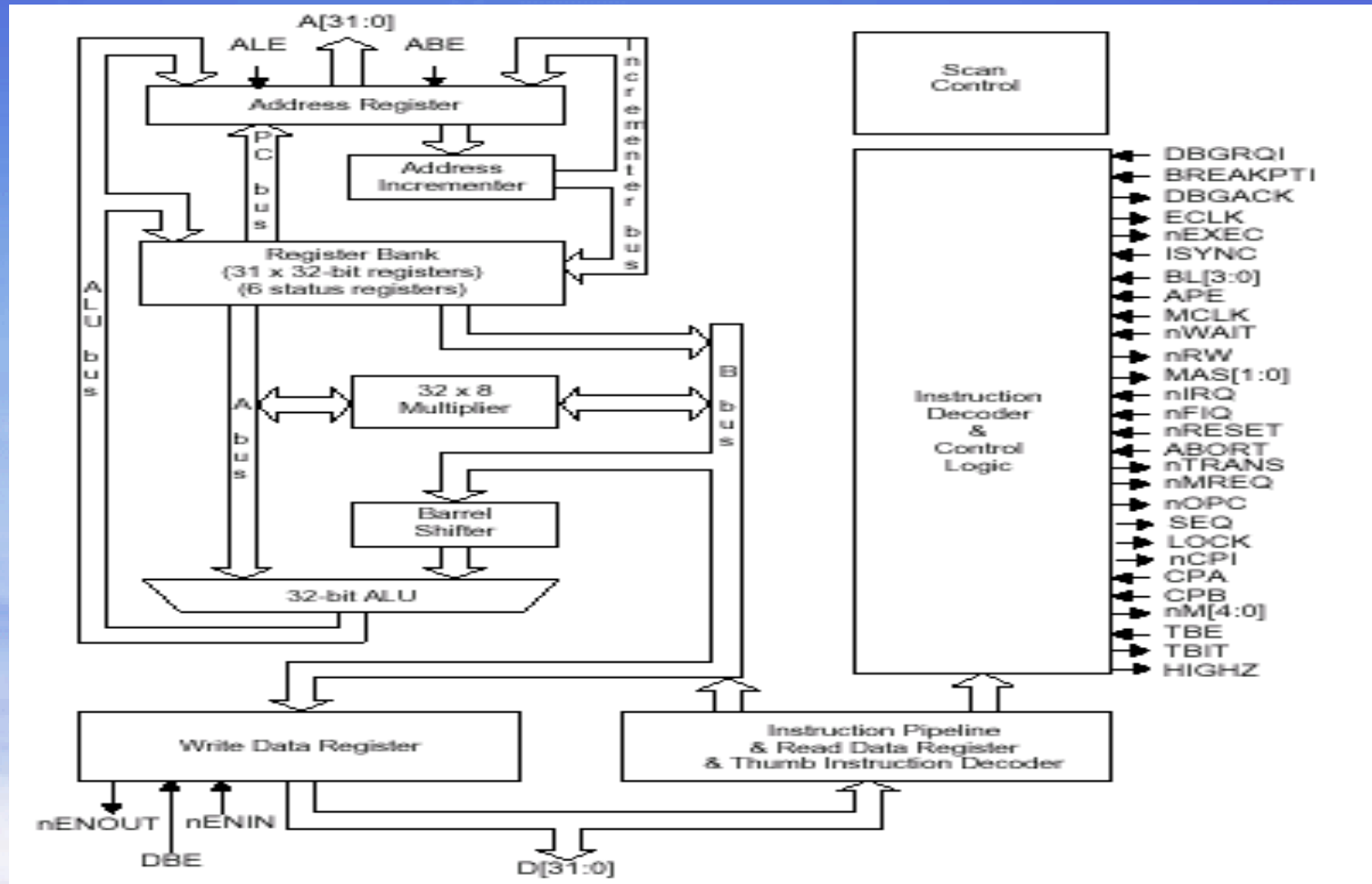
◆ 深圳市英蓓特信息技术有限公司

<http://www.embedinfo.com>

主要内容

- ❖ 程序员模型
- ❖ MMU与协处理器
- ❖ ARM指令系统
- ❖ THUMB指令系统
- ❖ 调试接口
- ❖ 异常向量表

程序员模型



程序员模型

- ❖ 运行状态：

 - ARM 状态（32位）

 - THUMB 状态（16位）

- ❖ 状态切换

 - BX Rm

 - * 异常自动切换到ARM状态

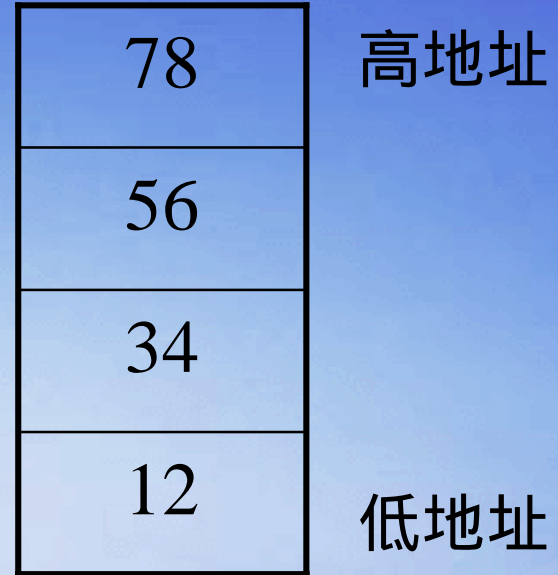
程序员模型

❖ 存储器格式

Big Endian format

高位字节存放在低地址

如：**0x12345678**在存储器中的存储顺序如图所示



程序员模型

❖ 存储器格式

Little Endian format

低位字节存放在低地址

如：**0x12345678**在存储器中的存储顺序如图所示



程序员模型

❖ 数据类型

字节	8位
半字	16位
字	32位

❖ 运行模式

用户、快速中断、中断、管理、中止、未定义

ARM寄存器组织

ARM处理器总共有37个寄存器：

- ❖ 31个32位通用寄存器
- ❖ 6个32位状态寄存器

ARM寄存器分类

❖ 通用寄存器

➤ 不分组寄存器 **R0~R7**

➤ 分组寄存器 **R8~R14**

(访问的具体物理寄存器取决于当前处理器模式)

➤ 程序计数器(PC) **R15**

❖ 程序状态寄存器 CPSR

(每种处理器模式都有单独的当前程序状态寄存器)

程序员模型

R0	R0	R0	R0	R0
R1	R1	R1	R1	R1
...
R7	R7	R7	R7	R7
R8	用做堆栈寄存器SP，每种异常模式			
...	用做子程序链接寄存器LR，可保存			
R10	子函数返回地址			
R11	用做程序计数器PC			
R12	含条件码标志、中			
R13	断点、当前处理器模式以及其它状态			
R15(PC)	和控制信息。			
CPSR	CPSR	CPSR	CPSR	CPSR
	SPSR_fiq	SPSR_svc	SPSR_abt	SPSR_irq

ARM状态寄存器结构

General Registers and Program Counter Modes

User32	FIQ32	Supervisor32	Abort32	IRQ32	Undefined32
R0	R0	R0	R0	R0	R0
R1	R1	R1	R1	R1	R1
R2	R2	R2	R2	R2	R2
R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6
R7	R7	R7	R7	R7	R7
R8	R8_fiq	R8	R8	R8	R8
R9	R9_fiq	R9	R9	R9	R9
R10	R10_fiq	R10	R10	R10	R10
R11	R11_fiq	R11	R11	R11	R11
R12	R12_fiq	R12	R12	R12	R12
R13	R13_fiq	R13_svc	R13_abt	R13_irq	R13_und
R14	R14_fiq	R14_svc	R14_abt	R14_irq	R14_und
R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)

Program Status Registers

CPSR	CPSR	CPSR	CPSR	CPSR	CPSR
	SPSR_fiq	SPSR_svc	SPSR_abt	SPSR_irq	SPSR_und

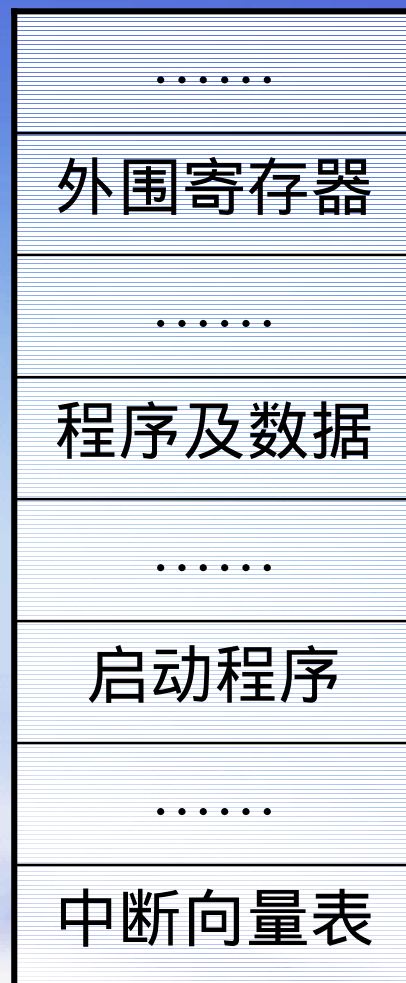
ARM的存储器组织

ARM的中断向量表分配在0x0到0x20，

ARM启动代码从0地址开始执行

复位的时候，0地址为ROM区

很多芯片允许存储区的重映射



0xFFFFFFF

0x00000000

MMU与协处理器

- ❖ 32位数据总线，32位地址总线
- ❖ 字、半字地址对齐
- ❖ ARM7TDMI核不包含MMU，由处理器外围提供
- ❖ 提供专用的协处理器指令

ARM指令集

- ❖ 位操作
- ❖ 条件执行
- ❖ 32 位 RISC 风格 (Reduced Instruction Set Computer)
- ❖ 数据总线 / 寄存器 均为32位宽
- ❖ 指令执行典型为一个时钟周期执行一条指令
- ❖ 多级指令流水线 (Pipeline) 结构: 3 for ARM 7, 5 for ARM 9

ARM 代码举例

ADDEQ R0, R1, R2 LSL #3 ;If the Z flag is set, then do a

;multiply R2 by 2^3 followed by an

;add of R1 and store result into R0

ARM指令集

- ❖ 寄存器装载和存储指令
- ❖ 算术和逻辑指令
- ❖ 移位操作指令
- ❖ 乘法指令
- ❖ 比较指令
- ❖ 分支指令
- ❖ SWI 指令
- ❖ 协处理器指令
- ❖ 伪指令

无下列指令：

- ❖ 除法指令
- ❖ IO指令
- ❖ 调用/返回
- ❖ 专门的堆栈指令

THUMB指令集

16 位 “Thumb” 指令集的代码量仅有ARM指令集的65% ,
而与16位存储器系统连接时 , 性能相当于ARM的160%

Thumb的局限性:

不支持乘法和累加指令.

条件跳转限制在 256 byte 偏移范围内,无条件跳转限制为
4K 偏移范围内 (而ARM为 32 Mbytes偏移)

很少指令为有条件的 (所有的ARM指令全为有条件的)

执行效率可能稍慢 , 典型情况下约为 10%

THUMB指令集

Thumb 代码举例

```
BNE    label           ;skip next two instructions if Z flag is 0
LSL    R2, R2, #3      ;multiply R2 by 2^3 and place in R2
ADD    R0, R1, R2      ;add R1+R2 and place in R0
:label           ;continue
```

Thumb指令集

- ❖ 存储器访问指令
- ❖ 数据处理指令
- ❖ 分支指令
- ❖ 中断和断点指令
- ❖ 伪指令

伪指令

- ❖ **ADR**
- ❖ **ADRL**
- ❖ **LDR**
- ❖ **ALIGN**
- ❖ **DCx**
- ❖ **EQUx**
- ❖ **OPT**

伪指令，它们不是处理器实际上能理解的指令，但可以转换成它能理解的某种东西。它们的存在能使你的程序更加简单。

调试接口

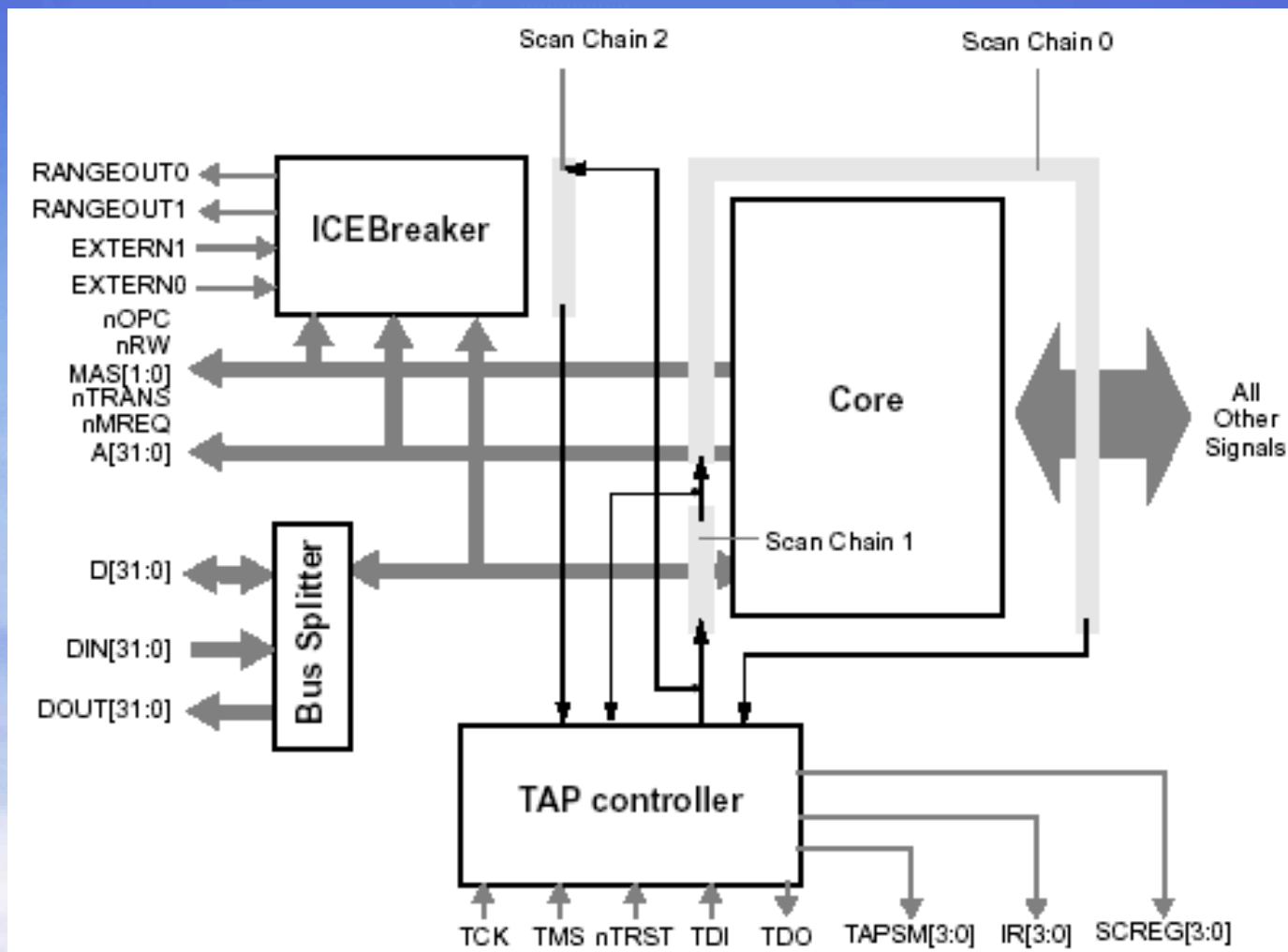
❖ JTAG调试接口

TMS、TCK、TDI、TDO、nTRST
ARM7TDMI没有提供输入信号的上拉

❖ 嵌入式ICE-RT逻辑

支持两个硬件断点，目标状态监控

ARM7TDMI 模块框图



异常向量

- ARM 的Exception Handler
 - Reset
 - Undefined instruction
 - SWI
 - Prefetch abort
 - Data abort
 - Reserved
 - FIQ
 - IRQ

RESET

- 正常情况下，系统reset后进入的入口
 - ENTRY主入口
 - 驻留于存储系统的0x0地址，占4Byte,机器码通常为EA0000XX
 - 通常在进入时将系统CPSR设为监控模式，退出时改为用户模式
 - 所有必须的BSP必须在此入口内完成
 - 使用汇编语言编写，可控制C代码入口

Undefined Instruction

- 未定义指令
 - 当程序员使用未定义指令时，系统出错处理的入口
 - 驻留于存储空间的0x4,4bytes
 - 处理程序通常做法是首先进行现场保护，然后Do nothing
 - 当出现此错误时，主因大致是仿真器跳入错误地址。

SWI

- 软中断
 - 驻留于0x8, 4Bytes
 - monitor程序的入口
 - 包含有丰富的指令寄存器
 - 适用于ARM和Thumb

Abort

- Perfect abort
 - 预取失败错误
 - 驻留于0x0c,4bytes
- Data abort
 - 取数据失败错误
 - 驻留于0x10,4bytes
 - 通常是保护现场，然后do nothing
 - 出错主因：程序跳飞（查程序）

FIQ和IRQ

- FIQ快速中断请求
 - 驻留于0x18,4bytes
- IRQ中断请求
 - 驻留于0x1c,4bytes
 - FIQ和IRQ处理原理相同
 - 所有的硬件中断源共用一个通道来进行IRQ或FIQ
 - 中断处理支持中断嵌套