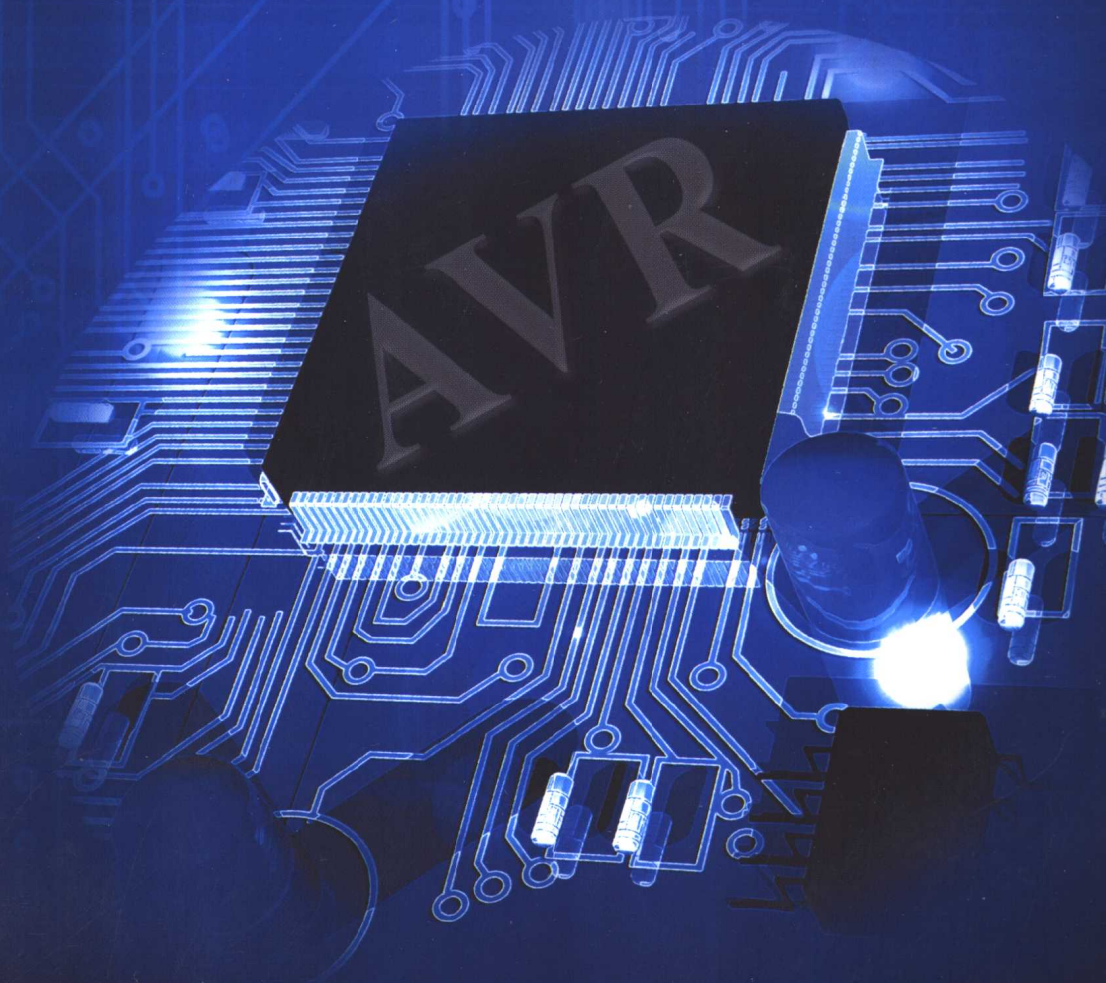


AVR 系列单片机

C 语言编程与应用实例

金春林 邱慧芳 张皆喜 编著



清华大学出版社

AVR 系列单片机 C 语言编程与应用实例

金春林 邱慧芳 张皆喜 编著

清华大学出版社
北 京

内 容 简 介

本书针对 Atmel 公司的 AVR 系列单片机和 ImageCraft 公司的 ICC AVR 开发环境,详细地介绍了 AT90LS8535 的 C 语言程序设计。全书共有 13 章,其内容既涉及到了单片机的结构原理、指令系统、内部资源和外部功能扩展,又包含了单片机的编程工具——ICC AVR C 编译器的数据类型、控制流、函数和指针等。

本书的特点是:深入浅出,从最基本的概念开始,循序渐进地讲解单片机的应用开发;列举了大量实例,使读者能从实际应用中掌握单片机的开发与应用技术。

本书适合作为从事单片机开发人员的参考用书。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

AVR 系列单片机 C 语言编程与应用实例/金春林,邱慧芳,张皆喜编著.—北京:清华大学出版社,2003
ISBN 7-302-07457-7

I .A… II.①金…②邱…③张… III.①单片微型计算机,AVR—程序设计②C 语言—程序设计
IV.①TP368.1②TP312

中国版本图书馆 CIP 数据核字(2003)第 094372 号

出 版 者:清华大学出版社

<http://www.tup.com.cn>

社 总 机:010-62770175

地 址:北京清华大学学研大厦

邮 编:100084

客户服务:010-62776969

组稿编辑:张 瑜

文稿编辑:刘 颖

封面设计:陈刘源

印 装 者:北京国马印刷厂

发 行 者:新华书店总店北京发行所

开 本:185×260 印 张:21 字 数:496 千字

版 次:2003 年 11 月第 1 版 2003 年 11 月第 1 次印刷

书 号:ISBN 7-302-07457-7/TP·5506

印 数:1~5000

定 价:30.00 元

前 言

自从 1976 年 Intel 公司推出 MCS-48 系列单片机以来,单片机技术得到了迅速的推广,已被广泛用于自动测量、智能仪表、工业控制及家用电器等各个方面。除了应用范围的扩大之外,单片机本身的发展也愈加迅速,如今,单片机的家族越来越庞大,系列产品越来越多,技术上也越来越先进。

AVR 系列单片机是 Atmel 公司生产的一种具有双总线结构的 RISC(Reduced Instruction Set Computer 的缩写,即精简指令系统计算机)单片机,它相对于传统的 CISC(Complicated Instruction Set Computer 的缩写,即复杂指令系统计算机)单片机而言,具有较短的指令周期和较快的运行速度。此外,AVR 单片机还具有片内 Flash、片内看门狗定时器、片内同步/异步串行接口、片内定时/计数器、片内模数转换器等多种内部资源。上述这些特点使采用 AVR 单片机的应用系统不仅具有运行速度快、结构简单、功能强大的特点,而且具有高可靠性和良好的经济性。

C 语言是一种编译型的结构化程序设计语言,具有简单的语法结构和强大的处理功能,具有运行速度快、编译效率高、移植性好和可读性强等多种优点,可以实现对系统硬件的直接操作。C 语言支持自顶向下的结构化程序设计方法,支持模块化程序设计结构。因此,用 C 语言来编写目标系统软件,可以大大缩短开发周期,且明显地增加软件的可读性,便于改进和扩充,从而开发出大规模、高性能的应用系统。

综上所述,以 AVR 单片机为基础,以 C 语言为工具开发单片机应用系统已成为单片机应用中的一个趋势。鉴于此,本书首先介绍了 AVR 单片机——AT90LS8535 的基本结构;然后介绍了 C 语言的数据类型、函数和指针等基础知识;最后结合大量实例,讲解了如何用 C 语言开发 AVR 系列单片机。因此,本书特别适用于单片机的初学者学习使用。

本书共 13 章。第一章为单片机简介;第 2 章为 AT90LS8535 单片机的基础知识;第 3 章介绍 AVR 单片机的 C 编程基础;第 4 章介绍 C 语言的数据结构;第 5 章介绍 C 的控制流;第 6 章介绍 C 的函数结构;第 7 章介绍 C 语言的指针类型;第 8 章介绍 C 的组合数据类型;第 9 章介绍 8535 的内部资源编程;第 10 章介绍 8535 的人机接口技术;第 11 章为 8535 单片机的扩展编程;第 12 章介绍 8535 的通讯接口;第 13 章介绍了系统设计中的程序处理方法。

由于作者的学识水平有限,书中难免有错误或不妥之处,恳请广大读者批评指正。

作者

2003 年 8 月于北京

目 录

第 1 章 单片机系统概述.....1	第 3 章 AT90LS8535 单片机的 C 编程..... 33
1.1 AVR 系列单片机的特点1	3.1 支持高级语言编程的 AVR 系列单片机 33
1.2 AT90 系列单片机简介3	3.2 AVR 的 C 编译器..... 34
第 2 章 AT90LS8535 单片机的 基础知识5	3.3 ICC AVR 介绍..... 35
2.1 AT90LS8535 单片机的总体结构.....5	3.3.1 安装 ICC AVR 35
2.1.1 AT90LS8535 单片机的 中央处理器7	3.3.2 设置 ICC AVR 37
2.1.2 AT90LS8535 单片机的 存储器组织8	3.4 用 ICC AVR 编写应用程序..... 39
2.1.3 AT90LS8535 单片机 的 I/O 接口9	3.5 下载程序文件..... 41
2.1.4 AT90LS8535 单片机 的内部资源10	第 4 章 数据类型、运算符和表达式..... 43
2.1.5 AT90LS8535 单片机 的时钟电路12	4.1 ICC AVR 支持的数据类型..... 43
2.1.6 AT90LS8535 单片机 的系统复位13	4.2 常量与变量..... 44
2.1.7 AT90LS8535 单片机 的节电方式14	4.2.1 常量 44
2.1.8 AT90LS8535 单片机 的芯片引脚15	4.2.2 变量 45
2.2 AT90LS8535 单片机的指令系统.....16	4.3 AT90LS8535 的存储空间 46
2.2.1 汇编指令格式20	4.4 算术和赋值运算..... 47
2.2.2 寻址方式20	4.4.1 算术运算符和算术表达式..... 47
2.2.3 伪指令22	4.4.2 赋值运算符和赋值表达式..... 49
2.2.4 指令类型及数据 操作方式24	4.5 逻辑运算..... 50
2.3 应用程序设计29	4.6 关系运算..... 51
2.3.1 程序设计方法29	4.7 位操作..... 52
2.3.2 应用程序举例30	4.7.1 位逻辑运算 53
	4.7.2 移位运算 54
	4.8 逗号运算..... 55
	第 5 章 控制流 57
	5.1 C 语言的结构化程序设计 57
	5.1.1 顺序结构 57
	5.1.2 选择结构 58
	5.1.3 循环结构 59
	5.2 选择语句..... 59

5.2.1 if 语句.....	59	7.3.4 指向多维数组的元素的 指针变量	106
5.2.2 switch 分支.....	62	7.4 字符串与指针.....	108
5.2.3 选择语句的嵌套	63	7.4.1 字符串的表示形式.....	108
5.3 循环语句	64	7.4.2 字符串指针变量与 字符数组的区别.....	111
5.3.1 while 语句	64	7.5 函数与指针.....	111
5.3.2 do...while 语句	65	7.5.1 函数指针变量.....	112
5.3.3 for 语句.....	66	7.5.2 指针型函数	113
5.3.4 循环语句嵌套	67	7.6 指向指针的指针	114
5.3.5 break 语句和 continue 语句	68	7.7 有关指针数据类型和运算小结	116
第 6 章 函数.....	71	7.7.1 有关指针的数据 类型的小结.....	116
6.1 函数的定义	71	7.7.2 指针运算的小结.....	116
6.1.1 函数的定义的一般形式.....	71	第 8 章 结构体和共用体.....	118
6.1.2 函数的参数	73	8.1 结构体的定义和引用	118
6.1.3 函数的值	75	8.1.1 结构体类型变量的定义.....	118
6.2 函数的调用	76	8.1.2 结构体类型变量的引用	120
6.2.1 函数的一般调用	76	8.2 结构类型的说明.....	121
6.2.2 函数的递归调用	79	8.3 结构体变量的初始化和赋值	122
6.2.3 函数的嵌套调用	82	8.3.1 结构体变量的初始化.....	122
6.3 变量的类型及其存储方式.....	83	8.3.2 结构体变量的赋值.....	123
6.3.1 局部变量	84	8.4 结构体数组.....	124
6.3.2 局部变量的存储方式	85	8.4.1 结构体数组的定义.....	124
6.3.3 全局变量	87	8.4.2 结构体数组的初始化.....	125
6.3.4 全局变量的存储方式	90	8.5 指向结构体类型变量的指针	127
6.4 内部函数和外部函数.....	92	8.5.1 指向结构体变量的指针	127
6.4.1 内部函数	92	8.5.2 指向结构体数组的指针	129
6.4.2 外部函数	92	8.5.3 指向结构体变量的指针 做函数参数.....	130
第 7 章 指针.....	94	8.6 共用体.....	131
7.1 指针和指针变量.....	94	8.6.1 共用体的定义.....	131
7.2 指针变量的定义和引用	95	8.6.2 共用体变量的引用.....	132
7.2.1 指针变量的定义	95	第 9 章 AT90LS8535 的内部资源.....	135
7.2.2 指针变量的引用	96	9.1 I/O 口	135
7.2.3 指针变量作为函数参数.....	97	9.1.1 端口 A	135
7.3 数组与指针	98	9.1.2 端口 B.....	136
7.3.1 指向数组元素的指针变量.....	98		
7.3.2 数组元素的引用 (通过指针).....	99		
7.3.3 数组名作为函数参数	101		

9.1.3	端口 C	141
9.1.4	端口 D	143
9.1.5	I/O 口的编程	148
9.2	中 断	148
9.2.1	单片机的中断功能	148
9.2.2	AT90LS8535 单片机的 中断系统	149
9.2.3	ICC AVR C 编译器的 中断操作	153
9.2.4	中断的编程	153
9.3	串行数据通信	155
9.3.1	数据通信基础	155
9.3.2	AT90LS8535 的同步 串行接口	156
9.3.3	AT90LS8535 的异步 串行接口	160
9.4	定时/计数器	167
9.4.1	定时/计数器的分频器	167
9.4.2	8 位定时/计数器 0	168
9.4.3	16 位定时/计数器 1	170
9.4.4	8 位定时/计数器 2	178
9.5	EEPROM	182
9.5.1	与 EEPROM 有关 的寄存器	182
9.5.2	EEPROM 读/写操作	183
9.5.3	EEPROM 的应用举例	184
9.6	模拟量输入接口	184
9.6.1	模数转换器的结构	185
9.6.2	ADC 的使用	185
9.6.3	与模数转换器有关 的寄存器	186
9.6.4	ADC 的噪声消除	188
9.6.5	ADC 的应用举例	188
9.7	模拟比较器	189
9.7.1	模拟比较器的结构	189
9.7.2	与模拟比较器有关 的寄存器	190
9.7.3	模拟比较器的应用举例	191

第 10 章 AT90LS8535 的人机

接口编程

10.1	键盘接口	193
10.1.1	非矩阵式键盘	193
10.1.2	矩阵式键盘	198
10.2	LED 显示输出	200
10.2.1	LED 的静态显示	202
10.2.2	LED 的动态扫描显示	203
10.2.3	动态扫描显示专用 芯片 MC14489	204
10.3	LCD 显示输出	208
10.3.1	字符型 LCD	208
10.3.2	点阵型 LCD	214
10.4	ISD2500 系列语音芯片的编程	224
10.4.1	ISD2500 的片内结构 和引脚	224
10.4.2	ISD2500 的操作	225
10.4.3	ISD2500 和单片机的 接口及编程	227
10.5	TP-μP 微型打印机	229
10.5.1	TP-μP 打印机的接口和 逻辑时序	229
10.5.2	P-μP 打印机的打印命令 和字符代码	230
10.5.3	AT90LS8535 与 TP-μP 系列 打印机的接口及编程	230
10.6	IC 卡	232
10.6.1	IC 卡读写装置	232
10.6.2	IC 卡软件	233

第 11 章 AT90LS8535 的外围扩展

11.1	简单 I/O 扩展芯片	246
11.1.1	用 74LS377 扩展数据 输出接口	246
11.1.2	数据输入接口	247
11.2	模拟量输出	250
11.2.1	D/A 转换器简介	250
11.2.2	8 位数模转换器 DAC0832	252

11.2.3	8 位数模转换器与单片机的接口及编程	253	11.7.1	DS18B20 的引脚和内部结构	284
11.2.4	12 位数模转换器 DAC1230	255	11.7.2	DS18B20 的温度测量	286
11.2.5	12 位数模转换器与单片机的接口及编程	257	11.7.3	AT90LS8535 与 DS18B20 的接口与编程	287
11.3	可编程 I/O 扩展芯片 8255A	258	第 12 章	AT90LS8535 的通信编程	290
11.3.1	8255A 的引脚和内部结构	258	12.1	串口通信	290
11.3.2	8255A 的工作方式	260	12.1.1	异步串口 UART 通信	290
11.3.3	8255A 的控制字	263	12.1.2	同步串口 SPI 通信	292
11.3.4	AT90LS8535 和 8255A 的接口	264	12.2	I ² C 总线	293
11.4	带片内 RAM 的 I/O 扩展芯片 8155	266	12.2.1	I ² C 总线协议	293
11.4.1	8155 的引脚和内部结构	266	12.2.2	采用 AT90LS8535 的并行 I/O 口模拟 I ² C 总线	297
11.4.2	8155 的 I/O 口工作方式	268	12.3	CAN 总线	300
11.4.3	8155 的定时/计数器	269	12.3.1	CAN 总线的特点	300
11.4.4	8155 的命令和状态字	269	12.3.2	CAN 协议的信息格式	300
11.4.5	AT90LS8535 与 8155 的接口及编程	270	12.3.3	CAN 控制器 SJA1000	301
11.5	定时/计数器芯片 8253	272	12.3.4	AT90LS8535 与 SJA1000 的接口及编程	307
11.5.1	8253 的信号引脚和逻辑结构	273	12.4	AT90LS8535 单片机与 PC 的串行通信	311
11.5.2	8253 的工作方式	274	12.4.1	基于 VC++6.0 的 PC 串口通信	311
11.5.3	8253 的控制字	275	12.4.2	应用实例	316
11.5.4	AT90LS8535 与 8253 的接口及编程	276	第 13 章	系统设计中的程序处理方法	319
11.6	实时时钟芯片 DS1302	277	13.1	数字滤波处理	319
11.6.1	DS1302 的引脚和内部结构	277	13.1.1	平滑滤波	319
11.6.2	DS1302 的控制方式	278	13.1.2	中值滤波	320
11.6.3	AT90LS8535 与 DS1302 的接口与编程	279	13.1.3	程序判断滤波	321
11.7	数字温度传感器 DS18B20	283	13.2	非线性处理	321
			13.2.1	查表法	322
			13.2.2	线性插值法	323

第 1 章 单片机系统概述

单片机就是包括了中央处理器 CPU(Central Processor Unit)、随机存储器 RAM(Random Access Memory)、只读存储器 ROM(Read Only Memory)和各种输入/输出单元的单芯片微机系统。

单片微型计算机也称为单片机, 目前已被广泛地应用于自动测量、智能仪表、工业控制及家用电器等各个方面。自从 1976 年 9 月美国 Intel 公司的 MCS-48 单片机问世以来, 世界各大厂商已相继研制出了 60 多个系列、上千个品种的单片机产品。随着半导体集成技术的提高和发展, 单片机正向着高主频、低功耗、低成本、多接口等方向发展。

1.1 AVR 系列单片机的特点

AVR 单片机是 Atmel 公司 1997 年推出的 RISC 单片机。RISC(精简指令系统计算机)是相对于 CISC(复杂指令系统计算机)而言的。RISC 并非只是简单地减少指令, 而是通过使计算机的结构更加简单合理而提高运算速度的。RISC 优先选取使用频率最高的简单指令, 避免复杂指令; 并固定指令长度, 减少指令格式和寻址方式的种类, 从而缩短指令周期, 提高运行速度。由于 AVR 采用了 RISC 的这种结构, 使 AVR 系列的单片机都具备了 1MIPS/MHz(百万条指令每秒/兆赫兹)的高速处理能力。

AVR 单片机吸收了 DSP 双总线的特点, 采用 Harvard 总线结构, 因此单片机的程序存储器和数据存储器是分离的, 并且可对具有相同地址的程序存储器和数据存储器进行独立的寻址。

在 AVR 单片机中, CPU 执行当前指令时取出将要执行的下一条指令放入指令寄存器中, 从而可以避免传统 MCS51 系列单片机中多指令周期的出现。

传统的 MCS51 系列单片机所有的数据处理都是基于一个累加器的, 因此累加器与程序存储器、数据存储器之间的数据交换就成了单片机的瓶颈; 在 AVR 单片机中, 寄存器由 32 个通用工作寄存器组成, 并且任何一个寄存器都可以充当累加器, 从而有效地避免了累加器的瓶颈效应, 提高了系统的性能。

AVR 单片机具有良好的集成性能。AVR 系列的单片机都具备在线编程接口, 其中的 Mega 系列还具备 JTAG 仿真和下载功能; 都含有片内看门狗电路、片内程序 Flash、同步串行接口 SPI; 多数 AVR 单片机还内嵌了 AD 转换器、EEPROM、模拟比较器、PWM 定时计数器等多种功能; AVR 单片机的 I/O 接口还具有很强的驱动能力, 灌电流可直接驱动继电器、LED 等器件, 从而省去驱动电路, 节约系统成本。

AVR 单片机采用低功率、非挥发的 CMOS 工艺制造, 除具有低功耗、高密度的特点外, 还支持低电压的联机 Flash、EEPROM 写入功能。

AVR 单片机还支持 Basic、C 等高级语言编程。采用高级语言对单片机系统进行开发是单片机应用的发展趋势。对单片机用高级语言编程可很容易地实现系统的移植，并加快软件的开发过程。

AVR 单片机具有多个系列，包括 ATtiny、AT90 和 ATmega。每个系列又包括多个产品，它们在功能和存储器容量等方面有很大的不同，但基本结构和原理都类似，而且编程方法也相同。表 1.1 为 AVR 各类型单片机的功能表。

表 1.1 AVR 单片机功能表

	Flash (KB)	Vcc (min)	EEPROM (B)	SRAM (B)	SPI	Watchdog	Clock (MHz)	RTC	On chip Oscillator	Interrupts	External	UART	8-bit Timer	16-bit Timer	PWM	10-bit A/D Channels	Instructions	ISP	Self Program Memory	Analog Comparat
ATtiny11L	1	2.7				Y	2		Y	4	1		1				90	2		Y
ATtiny11	1	4.0				Y	6		Y	4	1		1				90	2		Y
ATtiny12	1	1.8	64			Y	1		YI	5	1		1				90	Y		Y
ATtiny12L	1	2.7	64			Y	4		YI	5	1		1				90	Y		Y
ATtiny12	1	4.0	64			Y	8		YI	5	1		1				90	Y		Y
ATtiny15L	1	2.7	64			Y	1.6		YI	8	1		2		1	4	90	Y		Y
ATtiny28	2	1.8				Y	1		YI	5	2		1				90	N		
ATtiny28L	2	2.7				Y	4		YI	5	2		1				90	N		
AT90S1200	1	2.7	64			Y	12		Y	3	1		1				89	Y		Y
AT90S2313	2	2.7	128	128		Y	10			10	2	1	1	1	1		120	Y		Y
AT90LS2323	2	2.7	128	128		Y	4			2	1		1				120	Y		
AT90S2323	2	4.0	128	128		Y	10			2	1		1				120	Y		
AT90LS2343	2	2.7	128	128		Y	4		Y	2	1		1				120	Y		
AT90S2343	2	4.0	128	128		Y	10		Y	2	1		1				120	Y		
AT90LS4433	4	2.7	256	128	1	Y	4			14	2	1	1	1	1	6	120	Y		Y
AT90S4433	4	4.0	256	128	1	Y	8			14	2	1	1	1	1	6	120	Y		Y
AT90S8515	8	2.7	512	512	1	Y	8			11	2	1	1	1	2		120	Y		Y
AT90LS8535	8	2.7	512	512	1	Y	4	Y		15	2	1	2	1	3	8	120	Y		Y
AT90S8535	8	4.0	512	512	1	Y	8	Y		15	2	1	2	1	3	8	120	Y		Y
ATmega161L	16	2.7	512	1 K	1	Y	4	Y		20	3	2	2	1	4		130	Y	Y	Y
ATmega161	16	4.0	512	1 K	1	Y	8	Y		20	3	2	2	1	4		130	Y	Y	Y
ATmega163L	16	2.7	512	1 K	1	Y	4	Y	YI	17	2	1	2	1	3	8	130	Y	Y	Y
ATmega163	16	4.0	512	1 K	1	Y	8	Y	YI	17	2	1	2	1	3	8	130	Y	Y	Y

续表

	Flash (KB)	Vcc (min)	EEPROM (B)	SRAM (B)	SPI	Watchdog	Clock (MHz)	RTC	On chip Oscillator	Interrupts	External Interrupts	UART	8-bit Timer	16-bit Timer	PWM	10-bit A/D Channels	Instructions	Isp	Self Program Memory	Analog Comparat
ATmega323L	32	2.7	1	2 K	1	Y	4	Y	Y1	19	3	1	2	1	4	8	130	Y	Y	Y
ATmega323	32	4.0	1	2 K	1	Y	8	Y	Y1	19	3	1	2	1	4	8	130	Y	Y	Y
ATmega103L	128	2.7	4	4 K	1	Y	4	Y		16	8	1	2	1	4	8	121	Y		Y
ATmega103	128	4.0	4	4 K	1	Y	6	Y		16	8	1	2	1	4	8	121	Y		Y
ATmega8	8	4.5	512	1 K	1	Y	16	Y	Y1	18	2	1	2	1	3	8	130	Y	Y	Y
ATmega8L	8	2.7	512	1 K	1	Y	8	Y	Y1	18	2	1	2	1	3	8	130	Y	Y	Y
ATmega128	11	4.5	4	4 K	1	Y	16	Y	Y1		8	2	2	2	6	8	133	Y	Y	Y
ATmega128L	11	2.7	4	4 K	1	Y	8	Y	Y1		8	2	2	2	6	8	133	Y	Y	Y
ATmega16	16	4.5	512	1 K	1	Y	16	Y	Y	20	2	1	2	1	3	8	130	Y	Y	Y
ATmega16L	16	2.7	512	1 K	1	Y	8	Y	Y	20	2	1	2	1	3	8	130	Y	Y	Y
ATmega64	64	4.5	2	4 K	1	Y	16	Y	Y1		8	2	2	2	6	8	133	Y	Y	Y
ATmega64L	64	2.7	2	4 K	1	Y	8	Y	Y1		8	2	2	2	6	8	133	Y	Y	Y

① 高精度(5%)可编程内部 RC 振荡器。

② 编程时需给 RESET 管脚提供 12V 信号。

1.2 AT90 系列单片机简介

AT90 系列单片机是带片内 Flash 的在线可编程 8 位微控制器。

AT90 系列单片机是 AVR 单片机中的中档产品, 内部具有的 Flash 程序存储器的容量介于 ATtiny 和 ATmega 之间, 同时还具备较丰富的 I/O 接口, 是 AVR 单片机中性价比较高的一个系列。AT90 与 ATtiny 和 ATmega 系列的原理相同, 只是在功能上略有差异, AT90 系列的单片机的特点如下:

- 采用 120 条指令的 CPU 内核, 而 ATtiny 是 90 条, Atmega 是 130 条。
- 具有可在线编程的 Flash 存储器, 可反复擦写 1000 次以上。
- 片内含有少量字节的 EEPROM, 可以在线编程, 也可在程序运行期间由程序写入, 即便断电也不会丢失信息。
- 带有看门狗电路, 能保证在系统出错时, 程序有效复位。
- 具有较多的 I/O 接口, 且每个 I/O 接口都有很强的驱动能力, 能直接驱动继电器、LED 等器件。
- 带有片内的定时/计数器, 除实现定时、计数功能之外, 有些还具备比较匹配输出

和输入捕获功能。

- 多数单片机都含有同步串行接口 SPI 和异步串行接口 UART，便于实现系统与计算机或系统与系统间的通信。
- 部分单片机含有 AD 转换器，如 AT90S8535、AT90S4433，可直接对引脚输入的模拟电压信号进行转换。
- 部分产品具备 PWM(Pulse Width Modulation)输出功能，此输出既可作为控制信号使用，也可经滤波作为 D/A 转换器使用。
- 具有较多的中断源，便于程序的设计开发。
- 具有较为丰富的开发语言。除了汇编语言之外，还可以采用 Basic、C 等高级语言进行系统开发。
- 具有 DIP、TQFP 和 PLCC 多种封装形式。

第 2 章 AT90LS8535 单片机的基础知识

AT90 系列单片机是由 Atmel 公司开发的一系列基于 RISC 结构的 8 位微控制器。其中的 AT90LS8535 是这一系列中功能最全的产品，它不仅包含了通用的 I/O 功能，而且还具备一个 10 位的 AD 转换器，一个全双工的串行接口，一个具有计时功能的实时时钟和 3 个可编程的多功能定时/计数器。本章将通过对 AT90LS8535 单片机结构的叙述，来说明 AVR 单片机的系统结构，为后面章节的学习打好基础。

2.1 AT90LS8535 单片机的总体结构

AT90LS8535 单片机的总体结构如图 2.1 所示。各主要组成部分介绍如下：

- 算术逻辑运算单元 ALU。算术逻辑单元 ALU 能对数据进行算术运算和逻辑运算，是单片机的核心。ALU 只能运算，运算的操作数必须预先放在 32 个通用寄存器中，而运算的结果可以送到任何一个可寻址单元中。ALU 在完成运算后还会将本次运算的特征信息送到状态寄存器中。
- 内部程序存储器。AT90LS8535 内部含有 8 KB 的程序 Flash。Flash 中可以存放程序，也可存放变量的初始化值和表格数据。AT90LS8535 的程序存储器可通过编程逻辑单元的控制多次改写。
- 内部数据存储器。AT90LS8535 具有 512B 的内部数据存储器单元，可用于存放变量数据。
- 定时及控制单元。这部分确定了系统的时钟和复位逻辑。AT90LS8535 既可以采用芯片内 RC 振荡器提供的时钟信号，也可以采用外部时钟电路提供的时钟信号。AT90LS8535 的复位逻辑支持 3 种复位源——上电复位、外部复位和看门狗复位。
- 并行 I/O 接口。AT90LS8535 具有 4 个 8 位的 I/O 接口(PA、PB、PC、PD)。各 I/O 接口可设置成输入口，也可设置成输出口。当设置成输出口时，输出缓冲器能吸收约 20 mA 的电流，可直接驱动 LED、继电器等部件。复位后，各 I/O 接口均为高阻状态。
- 中断单元。AT90LS8535 有 16 个中断源，每个中断源都有独立的中断程序入口地址，所有的中断事件都有自己的使能位，可以根据使用需要屏蔽或使能。
- 模数转换器。AT90LS8535 具有 8 通道 10 位精度的逐次逼近式模数转换器。AD 转换部分通过 8 通道模拟转换器与 A 口的 8 个输入端相连，从而实现 8 通道的 AD 转换功能。
- 定时/计数器。AT90LS8535 有两个可分频的 8 位的定时/计数器和一个可分频的 16 位定时/计数器。其中的 1 个 8 位定时/计数器具有比较输出和 PWM 功能；而

16 位的定时/计数器不仅具有比较输出和 PWM 功能，还可实现输入捕获。

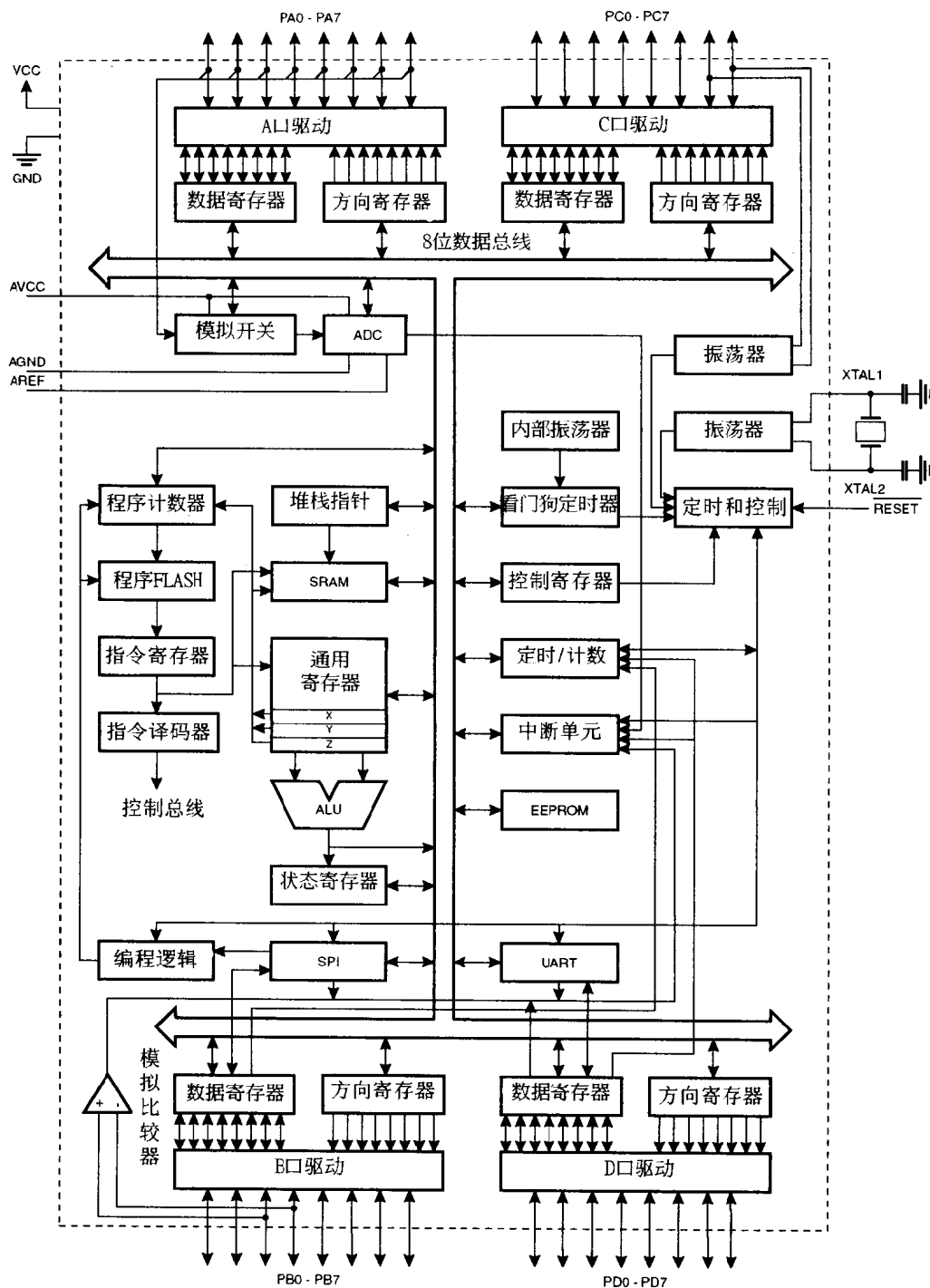


图 2.1 AT90LS8535 单片机的总体结构

2.1.1 AT90LS8535 单片机的中央处理器

AT90LS8535 的中央处理器包括 32 个通用寄存器、一个算术逻辑运算单元和相应的控制逻辑部分。

32 个 8 位的通用寄存器可实现单周期访问，其中的 6 个寄存器可以构成 3 个 16 位的用于数据寻址的指针 X、Y、Z，以提高 ALU 的地址运算能力。

AT90LS8535 同所有的 AVR 单片机一样，都采用了程序和数据总线分离的 HARVARD 结构。这种结构可以保证 ALU 在执行当前指令的同时去取下一条指令，从而使绝大多数指令在一个时钟内完成。

AT90LS8535 单片机的中央处理器结构如图 2.2 所示。

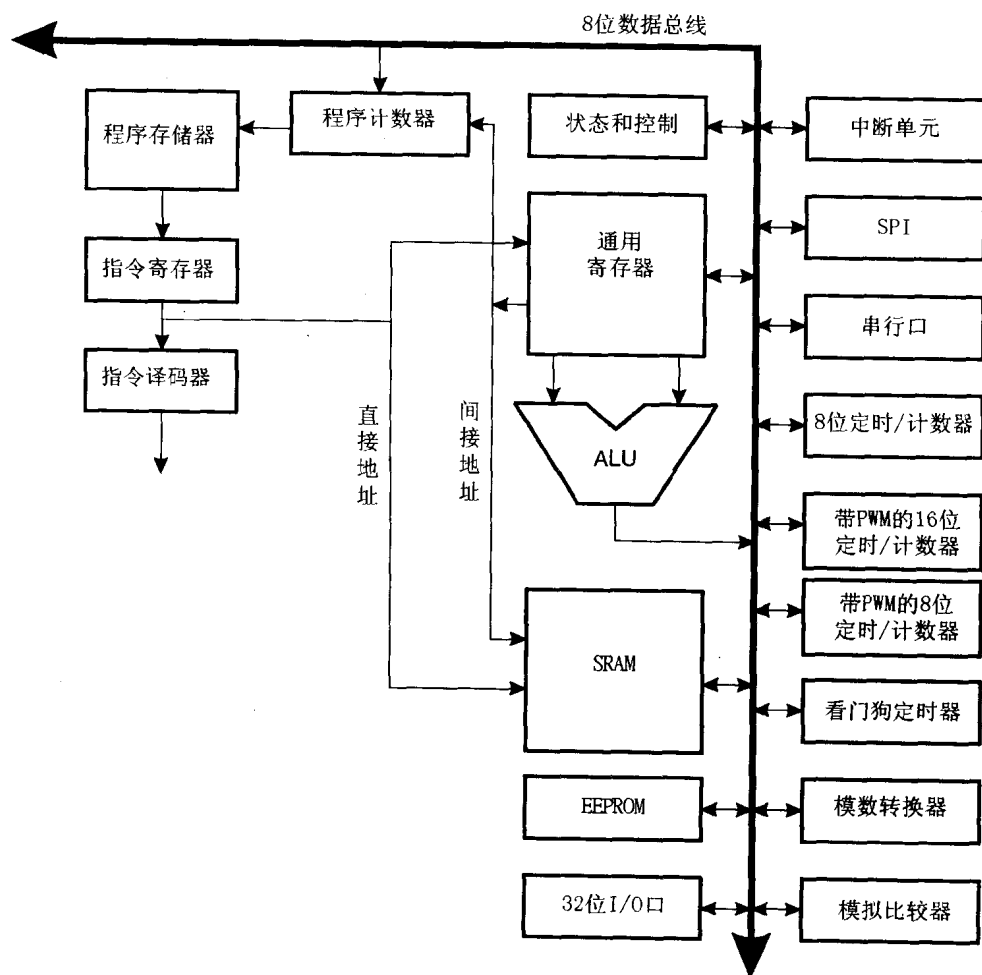


图 2.2 AT90LS8535 单片机的中央处理器结构

- 512 B 的 SRAM 用于存放程序中的变量，并且在中断或子程序调用时生成堆栈，以便中断或子程序返回后，系统能返回到复位前的状态。

- 中断单元由各中断控制器和状态寄存器的中断使能位构成，每个中断源都有一个独立的中断矢量入口地址。各中断源在相应使能位为 1，且全局中断使能位为 1 的时候，产生中断请求。
- ALU 是算术和逻辑运算的核心。它以 32 个通用寄存器中的值为操作数，并把运算的结果送到预定的寄存器或存储器中。

2.1.2 AT90LS8535 单片机的存储器组织

AT90LS8535 单片机的存储器包括程序 Flash、片内 EEPROM 和片内数据存储器 3 个部分，这 3 个部分都具有独立的寻址机构和寻址方式。因此对于重复的地址段，系统通过不同的寻址指令来加以区分。

AT90LS8535 具有 8KB 的程序 FLASH，它可通过在线编程的方式擦除或改写。Flash 中的某些特定的单元是保留出来供系统使用的，如 \$000 是系统复位后的入口地址，\$001~\$010 是中断服务程序的入口地址，因此用户所编写的程序不应进入这些区域。

AT90LS8535 含有 512B 的片内 EEPROM，用户程序可通过对其相关的控制寄存器、地址寄存器和数据寄存器的访问实现对 EEPROM 的读写。EEPROM 可实现至少 100 000 次的无故障擦写循环。

AT90LS8535 的片内数据存储器实际上包括通用寄存器、I/O 寄存器和 SRAM 这 3 个部分，但这 3 个部分是统一编址的，因此片内的数据存储器总共就占有 608 个地址。其中，通用寄存器的地址为 \$0000~\$001F，I/O 寄存器的地址为 \$0020~\$005F，内部 SRAM 的地址为 \$0060~\$025F。AT90LS8535 的 I/O 寄存器分布如表 2.1 所示。

表 2.1 AT90LS8535 的 I/O 寄存器分布表

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
\$3F (\$5F)	SREG	I	T	H	S	V	N	Z	C
\$3E (\$5E)	SPH	-	-	-	-	-	-	SP9	SP8
\$3D (\$5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0
\$3C (\$5C)	Reserved								
\$3B (\$5B)	GI_MSK	INT1	INT0	-	-	-	-	-	-
\$3A (\$5A)	GIFR	INTF1	INTF0						
\$39 (\$59)	TIMSK	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	-	TOIE0
\$38 (\$58)	TIFR	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	-	TOV0
\$37 (\$57)	Reserved								
\$36 (\$56)	Reserved								
\$35 (\$55)	MCUCR	-	SE	SM1	SM0	ISC11	ISC10	ISC01	ISC00
\$34 (\$54)	MCUSR	-	-	-	-	-	-	EXTRF	PORF
\$33 (\$53)	TCCR0	-	-	-	-	-	CS02	CS01	CS00
\$32 (\$52)	TCNT0	Timer/Counter0 (8 Bits)							
\$31 (\$51)	Reserved								
\$30 (\$50)	Reserved								
\$2F (\$4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	-	-	PWM11	PWM10
\$2E (\$4E)	TCCR1B	ICNC1	ICES1	-	-	CTC1	CS12	CS11	CS10
\$2D (\$4D)	TCNT1H	Timer/Counter1 – Counter Register High Byte							
\$2C (\$4C)	TCNT1L	Timer/Counter1 – Counter Register Low Byte							
\$2B (\$4B)	OCR1AH	Timer/Counter1 – Output Compare Register A High Byte							
\$2A (\$4A)	OCR1AL	Timer/Counter1 – Output Compare Register A Low Byte							
\$29 (\$49)	OCR1BH	Timer/Counter1 – Output Compare Register B High Byte							
\$28 (\$48)	OCR1BL	Timer/Counter1 – Output Compare Register B Low Byte							
\$27 (\$47)	ICR1H	Timer/Counter1 – Input Capture Register High Byte							
\$26 (\$46)	ICR1L	Timer/Counter1 – Input Capture Register Low Byte							
\$25 (\$45)	TCCR2	-	PWM2	COM21	COM20	CTC2	CS22	CS21	CS20

续表

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
\$24 (\$44)	TCNT2	Timer Counter2 (8 Bits)							
\$23 (\$43)	OCR2	Timer Counter2 Output Compare Register							
\$22 (\$42)	ASSR	-	-	-	-	AS2	TCN2UB	OCR2UB	TCR2UB
\$21 (\$41)	WDTCSR	-	-	-	WDTOE	WDE	WDP2	WDP1	WDP0
\$20 (\$40)	Reserved								
\$1F (\$3F)	EEARH								EEAR8
\$1E (\$3E)	EEARL	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0
\$1D (\$3D)	EEDR	EEPROM Data Register							
\$1C (\$3C)	EECR	-	-	-	-	EERIE	EENWE	EWE	EERE
\$1B (\$3B)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0
\$1A (\$3A)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0
\$19 (\$39)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0
\$18 (\$38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0
\$17 (\$37)	DDRB	ddb7	ddb6	ddb5	ddb4	ddb3	ddb2	ddb1	ddb0
\$16 (\$36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0
\$15 (\$35)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0
\$14 (\$34)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0
\$13 (\$33)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0
\$12 (\$32)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0
\$11 (\$31)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0
\$10 (\$30)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0
\$0F (\$2F)	SPDR	SPI Data Register							
\$0E (\$2E)	SPSR	SPIF	WCOL	-	-	-	-	-	-
\$0D (\$2D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0
\$0C (\$2C)	UDR	UART I/O Data Register							
\$0B (\$2B)	USR	RXC	TXC	UDRE	FE	OR	-	-	-
\$0A (\$2A)	UCR	RXCIE	TXCIE	UDRIE	RXEN	TXEN	CHR9	RXB8	TXB8
\$09 (\$29)	UBRR	UART Baud Rate Register							
\$08 (\$28)	ACSR	ACD	-	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0
\$07 (\$27)	ADMUX	-	-	-	-	-	MUX2	MUX1	MUX0
\$06 (\$26)	ADCSR	ADEN	ADSC	ADFR	ADIF	ADIE	ADPS2	ADPS1	ADPS0
\$05 (\$25)	ADCH	-	-	-	-	-	-	ADC9	ADC8
\$04 (\$24)	ADCL	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0
\$03 (\$20)	Reserved								
\$02 (\$22)	Reserved								
\$01 (\$21)	Reserved								
\$00 (\$20)	Reserved								

2.1.3 AT90LS8535 单片机的 I/O 接口

AT90LS8535 单片机具有 4 个 8 位的并行 I/O 接口，分别为 PA、PB、PC 和 PD。实际上它们都被归入专用的 I/O 寄存器之列，并可像访问通用寄存器一样进行寻址。在作为普通的数字 I/O 使用时，这 32 个 I/O 引脚的结构和特性是相同的，它们都可以通过相关控制寄存器定义为输入或输出口。

AT90LS8535 的 I/O 控制寄存器包括数据方向寄存器(DDRA~DDRD)、数据寄存器(PORTA~PORTD)和输入引脚“寄存器”(PINA~PIND)。输入引脚“寄存器”并不是真正意义上的寄存器，它只是一个可寻址的地址，该地址允许单片机直接读取 I/O 引脚上的逻辑值。

I/O 接口的口线逻辑电路如图 2.3 所示(以 PA 口为例)。整个电路由两个数据寄存器、3 个三态数据缓冲器和相应的输入输出元件构成。

当 A 口的方向寄存器 DDA 的第 n 位置为 1 时，PAn 定义为输出。由电路的逻辑原理可知：DDAn 的输出 Q 为 1 时，上拉 MOS 管截止，PORTAn 的三态门导通，从而使 PAn 的电平取决于 PORTAn 的输出 Q。

当 A 口的方向寄存器 DDA 的第 n 位置为 0 时，PAn 定义为输入。当 PORTAn 也为 0