

ATT7023 应用手册

版本：2.0
更新日期：2004/12/31

注意：珠海炬力集成电路设计有限公司保留对该手册在任何时候的改动权和解释权。请随时注意我公司的更新动态，该手册如有改动更新，我公司不另行通知。

Tel:0756-3392353

Fax:0756-3392251

[Http://www.actions.com.cn](http://www.actions.com.cn)

目录

1 芯片介绍	5
1.1 芯片特性	6
1.2 主要测量参数	7
1.3 电气参数	7
1.4 内部框图	9
1.5 引脚定义	11
1.6 应用示意图	13
2 兼容 8051 的 MCU 功能描述	14
2.1 8051 MCU 资源	15
2.2 程序控制部分功能	15
2.3 编程注意事项	16
3 电能计量	17
3.1 计量输出和计量原理	18
3.2 计量部分与 8051 数据交换	18
3.3 寄存器使用	18
3.4 数据寄存器使用方法	19
3.5 计度器输出控制	20
3.6 校表方法	21
3.7 启动和潜动	22
4 系统时钟	23
4.1 系统时钟介绍	24
4.2 系统时钟与工作模式	24
4.3 8051 时钟选择寄存器	24
5 LCD 部分	25
5.1 功能简述	26
5.2 编程指南	26
5.3 寄存器	28
6 LED 部分	29
6.1 功能简述	30

6.2 编程指南	31
7 UART 功能	32
7.1 功能简述	33
7.2 编程指南	33
7.3 寄存器	33
8 SPI 功能	34
8.1 功能简述	35
8.2 编程指南	35
8.3 寄存器	36
9 电源监控	37
9.1 功能简述	38
9.2 Standby Mode 和 Idle Mode	38
9.3 工作模式控制	40
9.4 电源供电系统	40
10 实时时钟和定时器控制部分	41
10.1 功能简述	42
10.2 RTC 部分编程指南	42
10.3 定时器部分编程指南	42
10.4 寄存器	42
11 通用 I/O 口	43
11.1 功能描述	44
11.2 寄存器	44
12 芯片封装	45
13 开发工具	46
13.1 软件开发	47
13.2 写码工具	47
13.3 FirmWare 保密设置	47
13.4 OTP ROM 出厂设置	47
14 应用线路图及 PCB 注意事项	48
附录 1 所有寄存器列表	50
附录 2 8051 相关资料	66

附录 3 [本应用手册更新内容](#) 70

1 芯片介绍

ATT7023 是一颗计量+MCU 的单相多功能电能表芯片，其内部包含 MCU、计量单元、时钟单元、电源管理单元、电压监测单元、看门狗单元、红外接口、异步串行接口、SPI 接口、32Kbyte OTP (One Time Programmable) 程序存储单元、256 字节掉电数据存储单元、2K 动态数据存储单元、两个定时器、通用 I/O 口、LCD/LED 数码管驱动单元。

所有的资源由内建的 MCU 统一管理，应用程序由用户自行开发，放在 32Kbyte 的 OTP ROM 中，通过访问相应的寄存器，用并行的方式传递数据，来控制所有的硬件资源。

内 容

页码

1.1	芯片特性.....	6
1.2	主要测量参数.....	7
1.3	电气参数.....	7
1.4	内部框图	9
1.5	引脚定义	11
1.6	应用示意图.....	13

1.1 芯片特性

计量部分

- 高准确度，根据 GB/T 17883 和 GB/T 17215（等效于 IEC687 和 IEC1036）设计，超过 1000:1 的动态范围内误差小于 0.1%；
- 能够输出电压有效值，电流有效值，线电压频率；
- 支持复费率电能表，有功功率平均值从 ATT7023 引脚 F1/F2 或 F3/F4 以频率方式输出，F1/F2 和 F3/F4 能直接驱动机电式计度器和两相步进电机；同时支持两相四拍电机，此时有功功率平均值从 FA1/FA2/FA3/FA4 或 FB1/FB2/FB3/FB4 输出；
- 有功电能和无功电能脉冲从引脚 CF1 和 CF2 以较高频率方式输出，用于仪表校验；
- 具有防窃电功能；
- 片内带有防潜动功能（空载阈值）；
- 片内基准电压 $2.5V \pm 8\%$ （温度系数典型值 $30\text{ppm}/^\circ\text{C}$ ）；
- 电流通道中的可编程增益放大器（PGA），能方便使用各种方式取样。

程序部分

- 内建 8051 兼容微处理器；
- MCU 直接寻址的 $2K+256\text{byte}$ SRAM；
- MCU 直接寻址的 32Kbyte 程序 OTP；
- 提供两个 16 位的定时、计数器，两个 8 位定时器；
- 提供 256 字节的低功耗 SRAM（Standby SRAM），系统掉电情况下，由后备电源维持数据不丢失；

外围电路

- 内建 UART 口，全双工：可应用于 485 远程抄表；
- 四线 SPI 串行通讯接口；
- 内建电源电压监测单元（Voltage Detector），根据系统电源及后备电源的情况，完成工作模式的转换及 MCU、外围电路的复位；
- 专门的 I/O 接口，可以驱动外接 EEPROM 或者其它外部设备；
- 专门的 $4\text{Common} \times 40\text{Segment}$ LCD 显示驱动电路；
- 1 个低功耗实时时钟单元（Real Time Clock），能自动判断闰年及每个月的天数。通过外接的 32.768KHz ，可以调整时间精度；
- 具有可编程时钟脉冲输出端，可编程输出 1、2、4、8、32、128、1024、 32768Hz ；
- 内建看门狗（WatchDog）电路；
- 内部产生 38K 的红外调制波；

电源

- +5V 单电源、低功耗。工作电流为 13mA ；
- 整体电路在电压低于 $3.0V \pm 5\%$ 时，进入休眠状态（RTC 部分除外）；
- 休眠状态的整体电路的耗电小于 $5\mu\text{A}$ 。

1.2 主要测量参数

ATT7023 主要测量参数:

电压有效值

电流有效值

线电压频率

有功功率、有功电能

无功功率、无功电能

典型应用:

单相电表

单相复费率电表

单相多功能电表

单相防窃电电表

1.3 电气参数

电特性 (Ta=25°C, AVCC=5V, VCC=5V, f_{osc}=6MHz)

测量项目	符号	测量条件	测量点	最小	典型	最大	单位
正电源电流	I _{dd}	V _v =0.248V V _i =1.75mV	电源输出		13		mA
Standby Mode 电流	I _{dd}		电源输出			5	uA
Idle Mode 电流	I _{dd}		电源输出			40	uA
参考电压	V _{ref}	V _v , V _i =0	Pin7	2.3	2.5	2.7	V
参考电压 温度系数	δ		Pin7		30	60	Ppm/°C
模拟输入			V1P V1N V2P V2N V3P V3N			±1	V
失调误差 通道1 通道2 通道3		G=1	V1P V1N V2P V2N V3P V3N			20 20 20	mV mV mV
增益误差 通道1 通道2					±4% ±4%		
增益匹配误差 通道1 通道2					±0.3% ±0.3%		

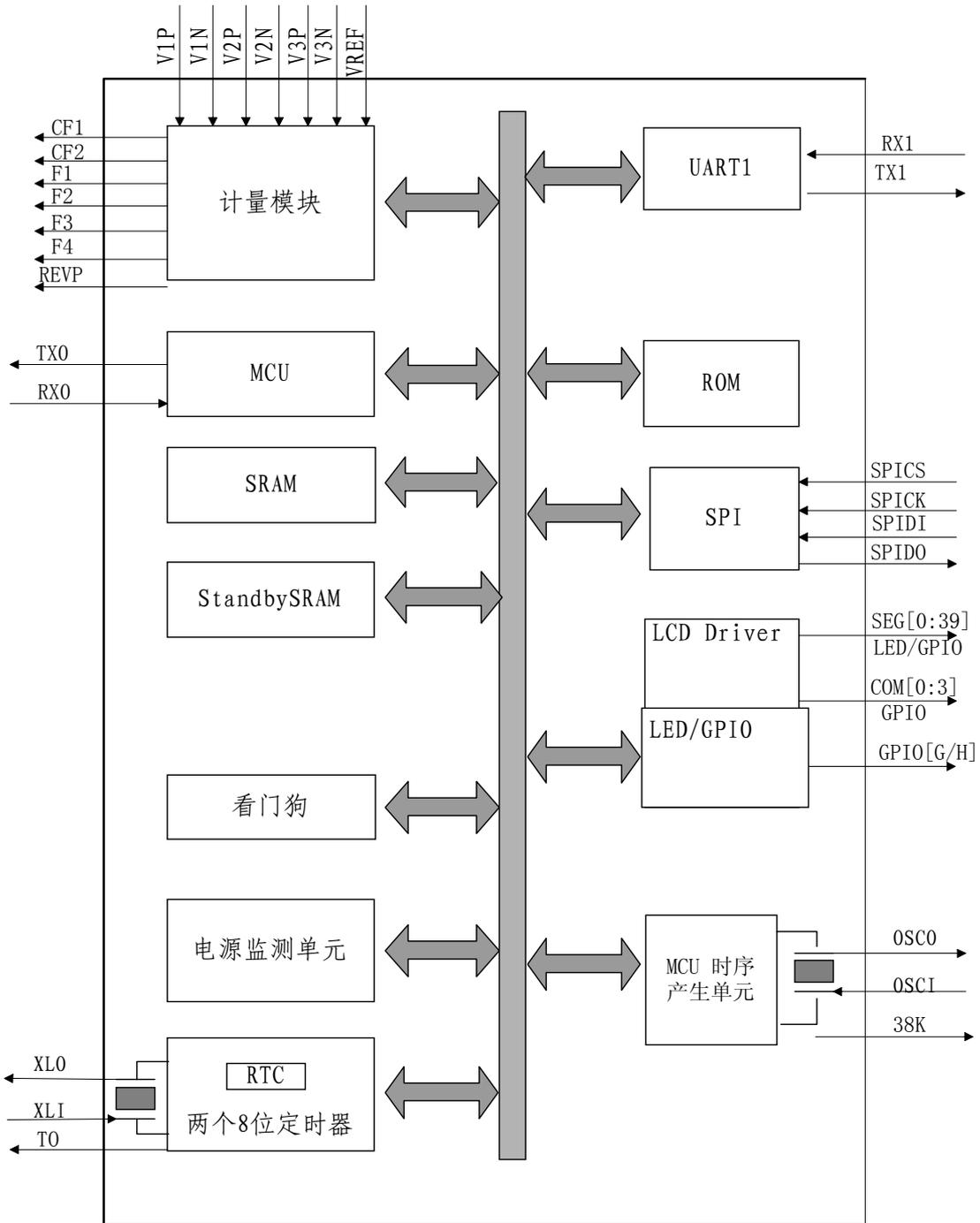
有功功率 误差%	e	动态范围 1000: 1 PF=1			±0.1%		
有功功率 误差%	e	动态范围 1000: 1 PF=0.5 频率 = 50Hz			±0.2 %		
有功功率 误差%	e	动态范围 1000: 1 PF=0.5 频率=45~65Hz				±0.5 %	
高频时钟输入					6		MHz
低频时钟输入					32768		Hz
有功, 无功功率 脉冲输出 CF1, CF2	V	V _v =0.248V (输入 V _i =1.75mV 条件) I _{OH} =5mA I _{OL} =5mA	Pin10, Pin11	4.5		0.4	V V
计数器驱动电 压 F1 F2 F3 F4	V	I _{OH} =10mA I _{OL} =10mA	Pin 95-98	4.5		0.4	V V
计数器驱动电 压 FA1/FA2/FA3/FA4 FB1/FB2/FB3/FB4	V	I _{OH} =10mA I _{OL} =10mA	Pin 86-89 91-94	4.5		0.4	V V
逻辑输入 输入高电平 输入低电平 输入电流 输入电容	V _{INH} V _{INL} I _{IN} C _{IN}		所有逻辑 输入	2.4		0.8 3 10	V V uA pF
逻辑输出 输出高电平 输出低电平	V _{OH} V _{OL}	I _{OH} =5mA I _{OL} =0.8mA	所有逻辑 输出	4		0.4	V V

极限参数 (T = 25 °C)

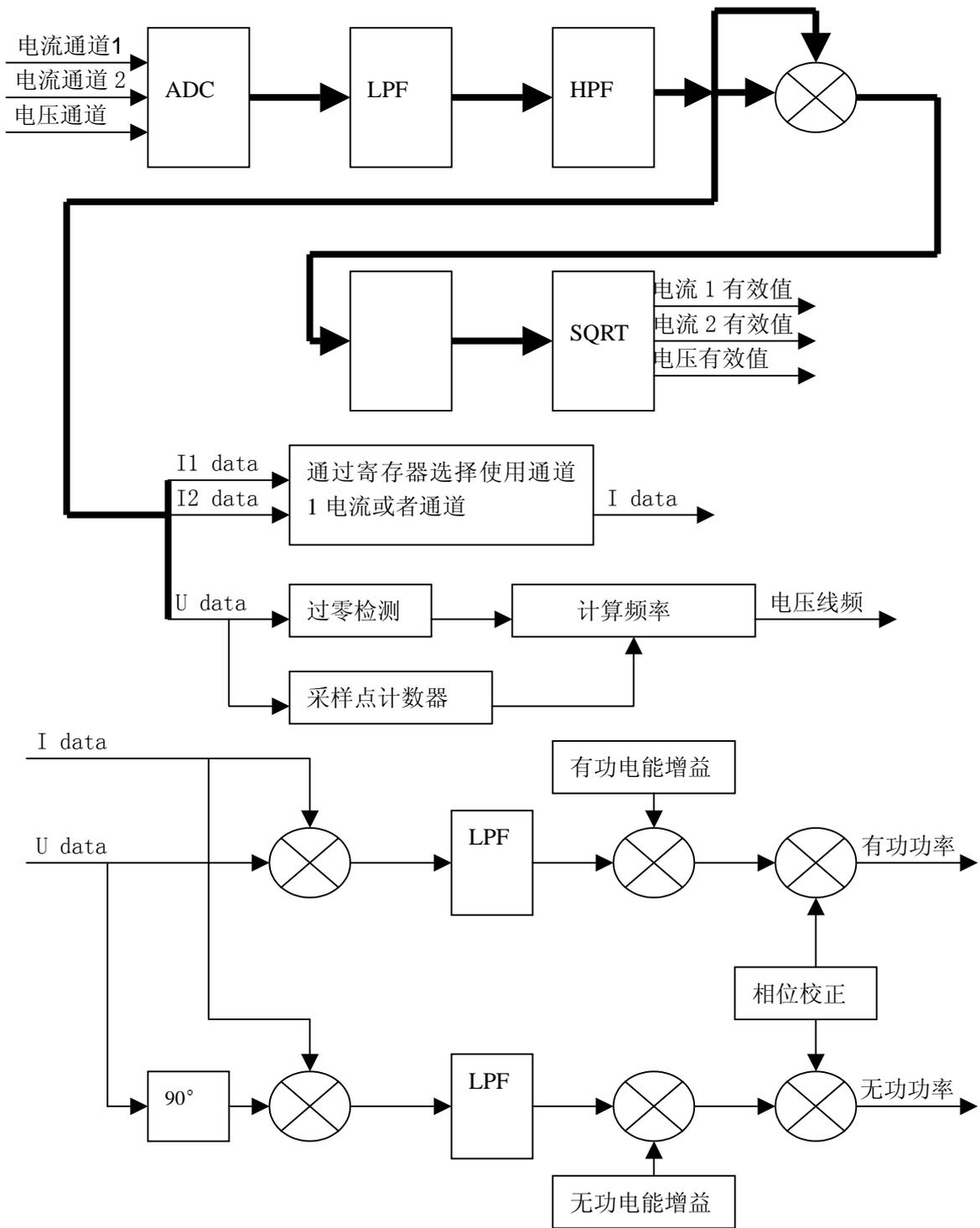
项 目	符 号	极 值	单 位
数字电源电压	VCC	-0.3 ~ +7	V
模拟电源电压	AVCC	-0.3 ~ +7	V
电流采样电压	V _v	-6 ~ +6	V
电压采样电压	V _i	-6 ~ +6	V
工作温度	T _{opr}	-40 ~ +85	°C
贮藏温度	T _{str}	-65 ~ +150	°C

1.4 内部框图

(1) 芯片内部模块图



(2) 计量模块内部模块图



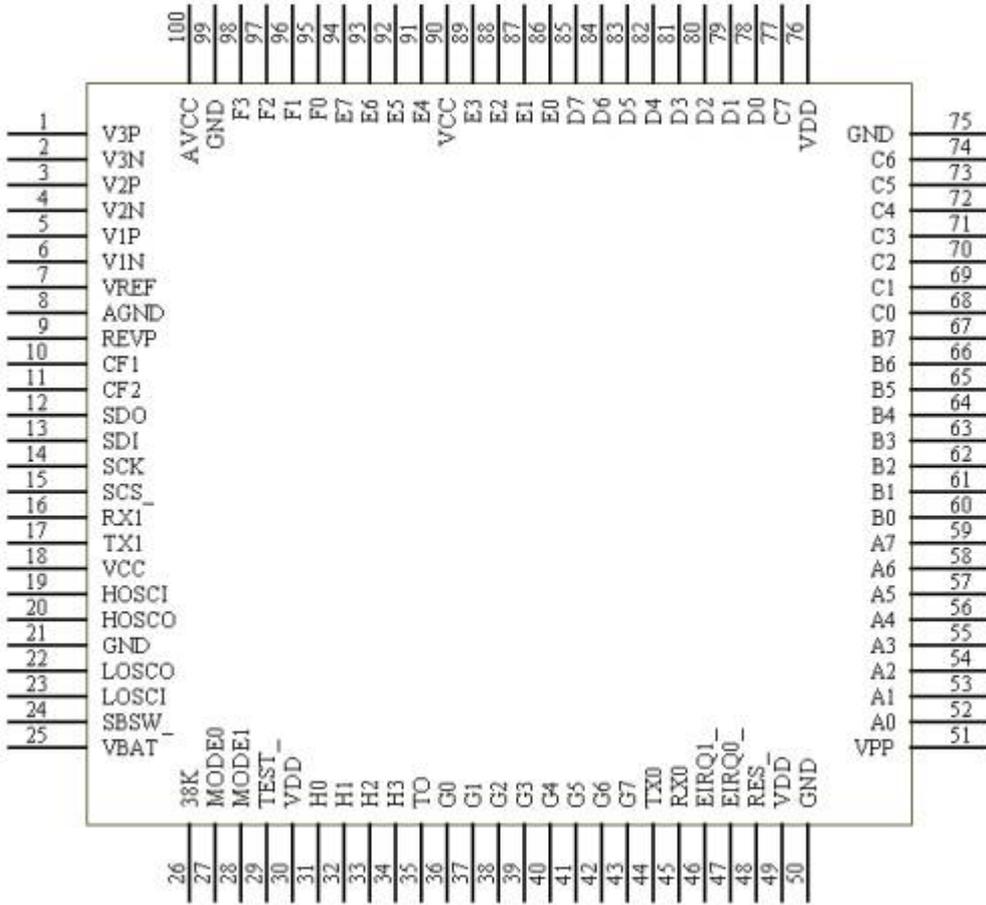
1.5 引脚定义

	Pin No.	I/O	Description
V3P	1	I	通道 3（电压通道）的正、负模拟输入引脚。完全差动输入，正常工作最大信号电平为 $\pm 660\text{mV}$ （峰峰值），内部都有 ESD 保护电路，能承受 $\pm 6\text{V}$ 的过电压，而不造成永久性损坏。
V3N	2	I	
V2P	3	I	通道 1 和通道 2（电流通道）的正、负模拟输入引脚。分别组成 V1P 与 V1N、V2P 与 V2N。完全差动输入方式，正常工作最大信号电平为 $\pm 470\text{mV}$ （峰峰值），内部都有 ESD 保护电路，能承受 $\pm 6\text{V}$ 的过电压，而不造成永久性损坏。
V2N	4	I	
V1P	5	I	
V1N	6	I	
VREF	7	I/O	参考电压，基准电压的输入、输出引脚。片内基准电压的标称值为 $2.5\text{V}\pm 8\%$ ，典型温度系数为 $30\text{ppm}/^\circ\text{C}$ 。
AGND	8	PWR	芯片模拟接地端。这是 ATT7023 模拟电路的接地参考点，该引脚应连接到 PCB 的模拟接地面。
REVP	9	O	反向有功功率标识输出，当检测到有功功率反向时输出高电平。
CF1	10	O	有功电能高频脉冲输出端，用于仪表校验。驱动电流为 5mA 。
CF2	11	O	无功电能高频脉冲输出端，用于仪表校验。驱动电流为 5mA 。
SPIDO	12	O	四线制 SPI 接口。输入 Pin 有 300K 上拉电阻到 VCC。
SPIDI	13	I	
SPICK	14	I	
SPICS_	15	I	
RX1	16	I	UART1 接收。内部接 300K 上拉电阻到 VCC。
TX1	17	O	由寄存器 883AH 的 bit6（SharpIRDis）控制： 若 SharpIRDis = 1，UART1 发送 若 SharpIRDis = 0，则 TX1 输出的是由红外 38K 调制过的 UART 信号。
VCC VDD	18/90 30/49/76	PWR	芯片数字电源输入。该引脚提供 ATT7023 数字电路的电源，正常工作电源电压应保持在 $5\text{V}\pm 5\%$ ，该引脚应使用 $10\mu\text{F}$ 电容并联 100nF 瓷介电容进行去耦。
HOSCI	19	I	系统时钟(6MHz)的输入端。
HOSCO	20	O	晶振的输出端。
GND	21/50 75/99	PWR	这是 ATT7023 数字电路的接地参考点，该引脚应连接到 PCB 的数字接地面。
LOSCO	22	O	32.768K 振荡器输出。
LOSCI	23	I	RTC 时钟，支持 32.768K 振荡器输入。
SBSW_	24	I	在 Standby Mode 不掉电，用来唤醒内部电路。
VBAT	25	PWR	3.6V 锂电池电源输入
38K	26	O	由 6M 时钟分频得到的 38K 红外调制波脉冲输出
MODE0 MODE1	27 28	I	模式控制，在正常应用时请参考 9.3 节接高或者接低。
TEST_	29	I	测试 PIN，正常应用时接高。
GPIOH0~3	31-34	I/O	通用 I/O 口
TO	35	O	由 RTC 时钟分频得到的时钟脉冲输出。
GPIOG0~7	36-43	I/O	通用 I/O 口
TX0	44	O	UART0 发送。
RX0	45	I	UART0 接收。内部接 300K 上拉电阻到 VDD。

EIRQ1_	46	I	外部中断，低电平有效。内部接 300K 上拉电阻到 VDD。
EIRQ0_	47	I	外部中断，低电平有效。内部接 300K 上拉电阻到 VDD。
RES_	48	I	复位信号输入，低有效。Schmitt Trigger，内部接 300K 上拉电阻到 VDD。
VPP	51	PWR	OTP 编程电压输入
SEG0~7/ NO0~NO7/ GPIOA	52~59	O	SEG0~7 驱动 PIN LED 扫描位 0~7 信号输出 GPIOA
SEG8~15/ A~P/ GPIOB	60~67	O	SEG 驱动 PIN LED 扫描段 A~P 信号输出。 GPIOB
SEG16~31/ GPIOC GPIOD	68~77 78~85	O I/O	SEG16~23 驱动 PIN。 GPIOC 和 GPIOD
FA1~FA4, FB1~FB4/ GPIOE/ SEG32~39	86~89 91~94	O I/O O	低频脉冲输出。用于驱动两相四拍计度器。驱动电流为 10mA。 通用 I/O 口。 SEG 驱动 Pin.
COM0~3/ F1~F4/ GPIOF0~3	95~98	O O I/O	4 COM 驱动 PIN。 低频脉冲输出。驱动两相两拍计度器。驱动电流 10mA。 通用 I/O 口。可以用作脉冲输出。
AVCC	100	PWR	芯片模拟电源输入。该引脚提供 ATT7023 模拟电路的电源，正常工作电源电压应保持在 5V±5%。

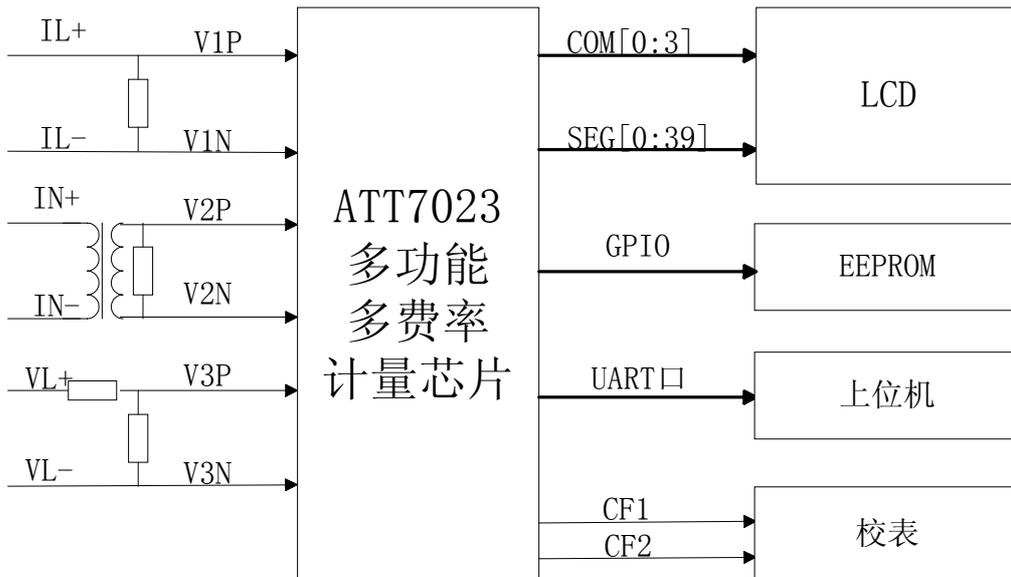
注：LCD、LED、计度器输出以及 GPIO 的复用情况具体见 LCD 部分寄存器部分说明。

管脚排列:



ATT7023 100 Pin LQFP Pin out Top view

1.6 应用示意图



2 兼容 8051 的 MCU 功能描述

ATT7023 内建的 MCU 采用 8 位的增强型 8051 微处理器，其特点是每个指令周期包含 4 个时钟周期，指令的执行速度比普通型 8051 平均快 2.5 倍。当时钟频率为 6MHz 时，执行一条单字节单周期指令平均需要的时间为 0.8 μ s。

其内部包含 128byte 的数据存储区，128byte 的特殊功能寄存器（SFR）区，2 个 16 位的定时器/计数器，一个全双工串行（UART）口，五个中断源。其指令系统兼容普通型 8051。

内 容

页码

2.1	8051 MCU 资源.....	15
2.2	程序控制部分.....	15
2.2.1	复位	15
2.2.2	中断.....	15
2.2.3	看门狗.....	15
2.3	编程注意事项.....	16

2.1 8051 MCU 资源

ATT7023内建的MCU采用8位的8051兼容微处理器。外部晶振使用6MHz，8051时钟通过884AH寄存器选择内建分频，分别为6M，3M，1.5M。默认的时钟频率是6MHz。

8051 的地址空间分配：

内部 SRAM: 128 byte (00H~7FH);

特殊功能寄存器 (SFR) 区: 128 byte (80H~FFH);

外部 SRAM: 2K (8000H ~ 87FFH);

寄存器区: 256 个字节, 占用地址空间 8800H ~ 88FFH;

Standby SRAM: 256 个字节, 占用地址空间 8900H ~ 89FFH。这部分 SRAM 在停电时由电池供电, 可以长期保存数据;

OTP ROM: 程序空间 32K, 地址从 0000H ~ 7FFFH。

8051 本身一个 UART 串行口作为 ATT7023 的串行口 0;

8051 本身提供两个定时/计时器 T0 和 T1。

8051 的复位、中断部分, 详见程序控制部分。

2.2 程序控制部分

2.2.1 复位:

复位以后, 8051 的程序从 ROM 空间 0000 地址开始运行。

2.2.2 中断:

8051 本身五个中断源分别为: 外部中断 0、定时/计数器 T0 溢出中断、外部中断 1、定时/计数器 T1 溢出中断、串行口中断。优先级从高到低。8051 有中断优先级寄存器: IP。

在 ATT7023 中, 将 8051 的两个外部中断源, 经过扩展后, 变成 11 个中断源。具体如下:

- 9 个内部中断作为 INTO 中断源;
- ATT7023 两个外部中断管脚 EIRQ0_、EIRQ1_作为 INT1 中断源。

当中断发生时, INTO 中断服务程序入口地址仍然是 0003H, INT1 中断服务程序入口地址仍然是 0013H, 由软件通过寄存器的状态位 IFn 便可以确定是哪一种中断, 从而进入相应的中断服务处理程序。

ATT7023 中, INTO 的中断源建议设定为电平触发方式, INT1 的中断源可以选择电平触发方式或者边沿触发方式。

中断源 T0、T1 和 UART0 在 8051 内有单独的 SFR, 因此不需要状态寄存器。他们的中断服务程序入口地址仍然是:

T0 中断:	000BH
T1 中断:	001BH
UART0 中断:	0023H

中断部分用到的寄存器请参考附录 1: 所有寄存器列表中 8844H~8847H 寄存器。

2.2.3 看门狗:

为了防止程序因为意外原因导致死机, ATT7023 专门设计了一套看门狗电路, 当程序死机后一定时间内 (由寄存器控制, 可选为 62.5 ms; 125 ms; 250 ms; 2s) 可以重新复位 8051。该电路是由一个计时器 (WDTimer) 来完成, WDTimer 的时钟源由 32768Hz 提供。

看门狗控制寄存器为 8849H, 详细说明见附录 1: 所有寄存器列表。

2.3 编程注意事项

ATT7023 内建的 8051 可以兼容一般的 8051，但是在应用上和一般的 8051 还有以下区别：

1. ATT7023 内部有两级中断控制，第一级在 8051 的内部（特殊功能寄存器 IE），第二级在 8051 的外部（寄存器 8846H 和 8847H），在程序中使用中断的时候，必须严格按照下列顺序来编写程序：先允许 8051 内部的中断，再允许 8051 外部的中断。
2. 4 cycles/instruction，而一般的 8051 是 12 cycles/instruction；
3. 有独立的数据、地址线，没有 P0~P3 口，因此在编程时不能使用特殊功能寄存器中 P0~P3 的地址：80H、90H、A0H、B0H；
4. ATT7023 内建的 8051 采用双数据指针来加强对外部数据快的访问，因此在一般 8051 的基础上，增加了特殊功能寄存器：DPL1 (84H), DPH1 (85H), DPS (86H)。DPTR0 的用法和一般 8051 的用法一样，如果要使用 DPTR1，则必须通过 DPS 来选择：当 DPS=00H，选择 DPTR0；当 DPS=01H，选择 DPTR1。在切换两个 DPTR 数据指针的时候，最快的方法是：INC 86H 或者 DEC 86H。需要注意的一点是：当程序中使用到 DPTR1 时，如果一般 8051 的仿真器没有 DPTR1 的功能，则无法仿真。
5. 特殊功能寄存器地址 92H 的内容用来代替使用语句 movx A, @Ri 和 movx @Ri, A 时 P2 的作用，当使用一般的 8051 仿真器开发程序时，如果用 P2 作高地址，可以使用上述两条语句，在将程序烧入 OTP ROM 时，必须将 P2(A0H)改为 92H。

6. 增加了特殊功能寄存器：CKCON(8EH)，其功能如下：

(8 位)	--	--	--	T1M	T0M	MD2	MD1	MD0
Reset	0	0	0	0	0	0	0	1

T1M: 8051 内部 Timer1 时钟选择。当 T1M=0, Timer1 时钟使用 clk/12 (于一般 8051 相同)；当 T1M=1, Timer1 时钟使用 clk/4。

T0M: 8051 内部 Timer0 时钟选择。与 T1M 类似。

MD2-0: 影响 MOVX 指令的执行时间，当设为 0-7 时 MOVX 指令的执行时间是 2-9 个指令周期。默认值为 1，因此 MOVX 指令的默认的执行时间是 3 个指令周期。

3 电能计量

内 容

页码

3.1 计量输出和计量原理.....	18
3.2 计量部分与 8051 数据交换.....	18
3.3 寄存器使用.....	18
3.4 数据寄存器使用方法	19
3.5 计度器输出控制.....	20
3.6 校表方法	21
3.6.1 校表模式介绍.....	21
3.6.2 校表寄存器.....	21
3.6.3 通信接口.....	22
3.7 启动和潜动.....	22

3.1 计量输出和计量原理

计量部分时钟采用主时钟的分频得到 1MHz，降 32 倍频到采样速率为 31.25KHz。

脉冲输出：

有功电能 CF1

无功电能 CF2

有功计度器输出 F1 F2/FA1 FA2 FA3 FA4

有功计度器输出 F3 F4/FB1 FB2 FB3 FB4

计量模块：

该模块完成电能计量。具体过程就是：实现电流、电压的采样，然后经过放大后，通过A/D转换器转换后成为数字信号，然后经过滤波、移相等数字信号的处理，得到电能信号。这样，电能信号可以被8051直接读取，并进行相关处理。计量电路可以完成有功/无功功率、电能值，以及电流、电压的有效值的求取。

计量电路分别采样两路电流信号，由软件控制，在计量模块内部选取第一路或者第二路电流与第三路电压的结果进行乘法处理，求取有功功率。上电时，默认第一路ADC采样作为电流通道输入。

3.2 计量部分与 8051 数据交换

计量模块与 8051 之间是并行通讯，主要通过寄存器以及中断进行控制。8051 可以直接访问计量模块内部寄存器，通过控制寄存器来控制通道增益、启动潜动、防窃电等功能，通过校表寄存器来进行软件校表，通过数据寄存器可以得到有效值、功率、能量、线电压频率等参数。

中断：

瞬时功率采样中断

有功电能寄存器已溢出(AEOF)

无功电能寄存器已溢出(QEOF)

参见后面中断寄存器部分，由计量部分发出中断请求，8051 响应后写相应的寄存器可将该中断清除。

工作过程如下：

- 8051 通过对控制寄存器写入数据，从而设定好工作模式，包括增益、采样频率、能量计量方式等。
- 8051 设定好中断控制。允许瞬时功率采样中断及其他必须的中断。
- 根据计量部分发出中断请求，8051 响应，读到数据，然后写相应的寄存器可将该中断清除。

3.3 寄存器使用

寄存器分控制寄存器、校表寄存器以及数据寄存器三类。

- 控制寄存器（88D0H、88D1H、88D4H）：控制计量部分的工作模式。
 - ADC 增益选择
 - 电流通道选择
 - 瞬时功率采样速率选择(1.75K/3.49K/6.99K/13.98K/27.97K)
 - CF 分频控制
 - 电能累加模式选择：
 - MODE1：绝对相加
 - MODE2：正负相加
 - 窃电标识/反向有功功率标识寄存器
 - 软件控制计量复位/启动

● 校表寄存器 (88D5H-88E3H): 8051 将校表数据写入其中对电能计量进行校准

- 有功电能增益(通道一) Gp1 16-bits
- 无功电能增益(通道一) Gq1 16-bits
- 相位校正(通道一) Gphs1 16-bits
- 有功电能增益(通道二) Gp2 16-bits
- 无功电能增益(通道二) Gq2 16-bits
- 相位校正(通道二) Gphs2 16-bits
- HF 高频输出 HF 16-bits
- F1/F2 低频输出 LF 8-bits

增益以及相位校正寄存器都是采用二进制补码形式, 最高一位为符号位, 低 15 位是小数位。

● 数据寄存器 (88E4H-88FEH): 传输计量数据到 8051

- 通道一电流有效值 Irms1 24-bits (Unsigned)
- 通道二电流有效值 Irms2 24-bits (Unsigned)
- 电压有效值 Urms 24-bits (Unsigned)
- 频率 Freq 16-bits (Unsigned)
- 有功电能 Ep 40-bits (Signed)
- 无功电能 Eq 40-bits (Signed)
- 有功功率波形 P 24-bits (Signed)
- 无功功率波形 Q 24-bits (Signed)

详细的寄存器说明请参见附录 1: 所有寄存器列表中第 2 部分: 计量模块寄存器。

3.4 数据寄存器使用方法

(1) 有效值

Vrms: 24 位数据, 补码形式

实际有效值为: $Urms = Vrms / 2^{23}$

例: 假如寄存器 88EAH-88ECH 值为 0x12ABCD (1223629), 则实际电压有效值为:

$$Urms = 1223629 / 2^{23} = 0.1458679 \text{ V}$$

注: 有效值不可能为负, 即最高位不可能为 1。

(2) 功率

X: 24 位数据, 补码形式

如果大于 2^{23} , 则 $XX = X - 2^{24}$

否则 $XX = X$

实际参数为: $PX = XX * 2^{23}$

实际功率为: $P = (18750 * OSO * PX) / (CF * HFConst)$

其中: OSO ---- 晶体频率: 6MHz

CF ---- 脉冲常数

HFConst ---- HF 分频寄存器 (88D5H-88D6H) 的值

例: 假如寄存器 88EDH-88EFH 值为 0x0419E2 (268770), 脉冲常数为 3200imp/kwh, HFConst 为 0x400 (1024), 则实际有功功率为:

$$P = (18750 * 6000000 * (268770 / 2^{23})) / (3200 * 1024) = 1100 \text{ W}$$

假如寄存器 88EDH-88EFH 值为 0XFBE61E (16508446)，其它条件同上，则实际参数为：

$$PX = (16508446 - 16777216) / 2^{23} = -0.03203988$$

实际有功功率为：

$$P = (18750 * 6000000 * (-0.03203988) / (3200 * 1024) = -1100 \text{ W}$$

(3) 线电压频率

Freq: 16 位数据

晶体频率为 0S0: 6MHz

实际频率为: $f=0S0/6/Freq$

例：假如寄存器 88FDH-88FEH 值为 0x4E20(20000)，则实际频率为：

$$f = 6000000/6/20000 = 50 \text{ Hz}$$

(4) 能量

40 位数据，补码形式

该参数与设定的脉冲常数有关。

如我们设定脉冲常数为 3200imp/kwh

则这个寄存器的单位为 1/3200kwh。

例：假如寄存器 88F3H-88F7H 值为 0x00ABCDEF00(2882400000)，脉冲常数为 3200imp/kwh，则实际有功能量为：

$$E_p = 2882400000/3200 = 900750 \text{ kw} \cdot \text{h}$$

注：当寄存器 88D0H 的 Pmode 和 Qmode 为 1 时，能量有可能出现负值，假如寄存器 88F3H-88F7H 值为 0xFFFFFFF0，脉冲常数为 3200imp/kwh，则实际有功能量为：

$$E_p = (0xFFFFFFF0 - 2^{40})/3200 = -0.01 \text{ kw} \cdot \text{h}$$

3.5 计度器输出控制

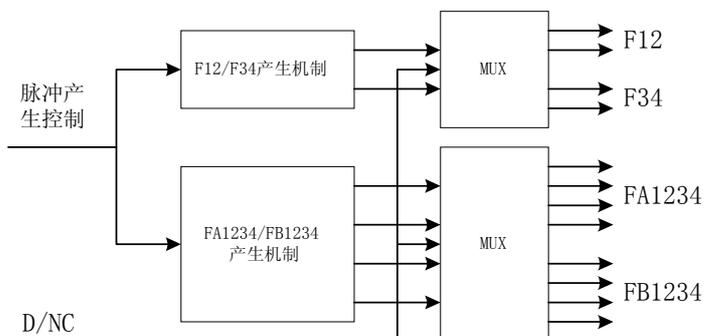
ATT7023 可以支持两相两拍和两相四拍两种计度器输出。

(1) 两相两拍计度器

可以支持两个两相两拍计度器输出，使用PIN为F1, F2, F3, F4, 其中F1/F2驱动一个计度器；F3/F4驱动另外一个计度器。管脚复用LCD显示的管脚COM0~COM3，计度器显示累计电能值。用一个控制位（寄存器884A的Bit3: D/NC），控制把脉冲输出到不同的两个计度器，控制信号可以由8051根据不同的时段要求——是高费率时段还是低费率时段，确定输出到不同的计度器。

D/NC: =0 脉冲只输出到 F1、F2。

=1 脉冲只输出到 F3、F4。



频率输出时序图

(2) 两相四拍计度器

对于两相四拍，ATT7023亦提供直接驱动两路计度器的脉冲，分别记录高低费率时段电能脉冲。使用PIN为FA1, FA2, FA3, FA4以及FB1, FB2, FB3, FB4。FA1/FA2/FA3/FA4驱动一个计度器，FB1/FB2/FB3/FB4驱动另外一个计度器。管脚复用LCD显示的管脚：PE0~PE7。

实现过程：由软件写显示方式寄存器BIT3(D/NC)，控制把脉冲输出到不同的两个计度器。

D/NC: =0 脉冲只输出到FA1、FA2、FA3、FA4。
 =1 脉冲只输出到FB1、FB2、FB3、FB4。

•884AH --- 显示方式寄存器

Read/Write

	LCD1	LCD0	SPC1	SPC0	D/NC	DisMode2	DisMode1	DisMode0
Reset	0	0	0	0	0	0	0	0

D/NC: 当用两相两拍计度器输出时，作为F1、F2和F3、F4切换的控制信号。当用两相四拍计度器输出时，作为FA1、FA2、FA3、FA4和FB1、FB2、FB3、FB4切换的控制信号。

0 两相两拍时，脉冲只输出到F1、F2；两相四拍时，脉冲只输出到FA1、FA2、FA3、FA4。

1 两相两拍时，脉冲只输出到F3、F4；两相四拍时，脉冲只输出到FB1、FB2、FB3、FB4。

3.6 校表方法

3.6.1 校表模式介绍

ATT7023可以通过两种方式来校表：简单模式以及软件模式

简单模式：通过改变外围的电阻值进行校表；

软件模式：通过8051将校表数据写入到相关的校表寄存器中。

3.6.2 校表寄存器

校表寄存器的详细说明请参见附录1：所有寄存器列表中88D5H~88E3H寄存器。

①高频输出参数 HF

参数HFConst决定校表用的高频脉冲输出CF的频率。

已知：

高频脉冲常数 N

低频脉冲常数 M

额定输入电压 U_n

额定输入电流 I_b

电压输入通道 V_u

电流输入通道 V_i

电流通道的增益 G

计算公式：

$$HF = INT \left[5625000000 \times \frac{2.396 \times V_u \times G \times V_i}{U_n \times I_b \times N} \right]$$

ATT7023有两个电流通道，此处的 V_i 表示正在使用的一路电流的电压信号，G表示正在使用的一路电流的增益（由88D0H寄存器决定）。

②设置低频输出参数 LF

LF=N/M

③有功电能增益校正 GP1、GP2

在 $1R(\cos(\Phi)=1)$ 时进行功率增益校正。

已知：

标准表上读出误差：err

计算公式：

$$P_{gain} = \frac{-err}{1+err}$$

如果 $P_{gain} \geq 0$ ，则 $GP = INT[P_{gain} * 2^{15}]$

否则 $P_{gain} < 0$ ，则 $GP = INT[2^{16} + P_{gain} * 2^{15}]$

④相位校正 GPhs1、GPhs2

在 $1R(\cos(\Phi)=1)$ 时功率增益已经校正好之后，再进行相位补偿。

通常只在 0.5L 处进行校正。

已知：

0.5L 处标准准备误差读数 err

相位补偿公式：

$$\theta = \arccos((1+err) * 0.5) - \pi / 3$$

如果 $\theta \geq 0$ ， $GPhs = \theta * 2^{15}$

否则 $\theta < 0$ ， $GPhs = 2^{16} + \theta * 2^{15}$

⑤无功电能增益校正 GQ1、GQ2

通常情况下，我们并不需要专门对无功增益进行校正，只要将已校正过的有功电能增益的值直接写入相应的无功电能增益寄存器中即可。如果一定要校正的话，可参照有功增益校正，方法类似。

3.6.2 通信接口

MCU 可以通过 UART 通讯口、SPI 通讯口等方式接收到校表数据，然后将校表数据写入 ATT7023 内部的校表寄存器中。

3.7 启动和潜动

ATT7023 的启动和潜动是通过判断功率的大小来实现的，具体就是软件判断功率大小，通过设定 88D1H 寄存器的停止位，然后由硬件完成启动、潜动的控制。

假如有功的启动电流设为 $0.2\%I_b$ ，则判断的功率为 $0.2\%P_b$ (P_b 为在 I_b 条件下校准后读出的有功功率寄存器的值，即为额定功率的值)，此时启动点为 $0.2\%P_b$ ，设定启动和潜动的方法就是在校准之后，读出有功功率寄存器 (P) 的值，和 $0.2\%P_b$ 相比较，如果 $P < 0.2\%P_b$ ，则写 88D1H 的 bit1 为 1，停止有功电能的计量，此时为潜动范围；如果 $P > 0.2\%P_b$ ，则写 88D1H 的 bit1 为 0，开始有功的计量，此时为启动。

无功的启动和潜动也是同样设定。

4 系统时钟

这部分主要介绍 ATT7023 的系统时钟以及内部模块在各种工作模式下的时钟状态。

内 容

页码

4.1 系统时钟介绍.....	24
4.2 系统时钟与工作模式	24
4.3 8051 时钟选择寄存器.....	24

4.1 系统时钟介绍

ATT7023 有两种工作时钟：高频晶振：6MHz，低频晶振：32768Hz。都需要通过外接石英晶体谐振器和相应的阻容参数来产生振荡。ATT7023 内部的计量模块始终用 6M 作为其工作时钟，RTC 模块、Watchdog、LCD 模块、LED 模块始终用 32768Hz 作为其工作时钟，而 8051 在 Normal Mode 下是用 6M/3M/1.5M 作为其工作时钟，在 Idle Mode 始终用 32768Hz 作为其工作时钟。

4.2 系统时钟与工作模式

ATT7023 有三种工作模式：Normal Mode，Standby Mode，Idle Mode。

在 Normal Mode:

计量模块工作，时钟用 6M

8051 工作时钟用 6M/3M/1.5M（根据 884AH 寄存器的 bit5, bit4 选择）

RTC 模块、Watchdog、LCD 模块、~~LED 模块~~工作时钟用 32768Hz

在 Standby Mode:

计量模块停止工作

8051 停止工作

Watchdog、LCD 模块、~~LED 模块~~停止工作

RTC 模块工作，时钟用 32768Hz

在 Idle Mode:

计量模块停止工作

8051 工作，时钟用 32768Hz

RTC 模块、LCD 模块工作时钟用 32768Hz

~~Watchdog、LED 模块停止工作~~

4.3 8051 时钟选择寄存器

在 Normal Mode，8051 的工作时钟可以选为 6M/3M/1.5M，根据 884AH 寄存器的 bit5, bit4 选择，默认选择 6M。

•**884AH** --- 显示方式寄存器

Read/Write

	LCD1	LCD0	SPC1	SPC0	D/NC	DisMode2	DisMode1	DisMode0
Reset	0	0	0	0	0	0	0	0

8051 的工作频率由 6M 的时钟频率分频而来。

SPC1	SPC0	
0	0	分频系数为 1 (6M)
0	1	分频系数为 2 (3M)
1	0	分频系数为 4 (1.5M)

其它位的说明请参见附录 1：所有寄存器列表。

5 LCD 部分

这部分主要介绍 ATT7023 的 LCD 驱动功能。

内 容

页码

5.1 功能简述.....	26
5.2 编程指南.....	26
5.3 寄存器.....	28

5.1 功能简述

ATT7023内建LCD驱动电路。有40个段信号输出管脚(SEG0-SEG39)和四个公共端输出管脚(COM0-COM3)，最多可支持4×40(共160段)LCD段码显示。支持 $V_{LCD}=3V$ 和 $5V$ 两种液晶以及4种frame frequency，通过控制寄存器进行选择。

通过显示模式控制寄存器884AH可以看到，ATT7023默认采用4COM × 24SEGMENT的LCD显示方式。

5.2 编程指南

在ATT7023中，驱动LCD显示所需要的数据寄存器为8810到8823。数据位为1时，SEG与COM对应的段就会点亮。使用之前，必须使用884A寄存器选择显示方式为LCD驱动，就是将884AH寄存器赋以正确的值；同时将8824H的bit0置1。

•8810H~8823H --- LCD 数据寄存器 Read/Write

Data		Bit7	Bit6	Bit5	Bit4		Bit3	Bit2	Bit1	Bit0
REG		COM0	COM1	COM2	COM3		COM0	COM1	COM2	COM3
8810H	SEG1					SEG0				
8811H	SEG3					SEG2				
8812H	SEG5					SEG4				
8813H	SEG7					SEG6				
8814H	SEG9					SEG8				
8815H	SEG11					SEG10				
8816H	SEG13					SEG12				
8817H	SEG15					SEG14				
8818H	SEG17					SEG16				
8819H	SEG19					SEG18				
881AH	SEG21					SEG20				
881BH	SEG23					SEG22				
881CH	SEG25					SEG24				
881DH	SEG27					SEG26				
881EH	SEG29					SEG28				
881FH	SEG31					SEG30				
8820H	SEG33					SEG32				
8821H	SEG35					SEG34				
8822H	SEG37					SEG36				
8823H	SEG39					SEG38				

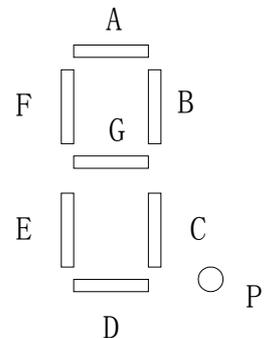
假如我们选取如右图(图)所示的七段码，选取下图(图)所示的玻璃，且玻璃显示的真值表见表 。

我们可以得到所需要的部分显示字库如下：

```

;CHAR:      0  1  2  3  4  5  6  7  8  9
LCD_CHAR:  DB 7BH, 60H, 3DH, 75H, 66H, 57H, 5FH, 70H, 7FH, 77H
;char:      0  1  2  3  4  5  6  7  8  9
LCD_CHAR_S: DB BEH, 0CH, D6H, 5EH, 6CH, 7AH, FAH, 0EH, FEH, 7EH

```



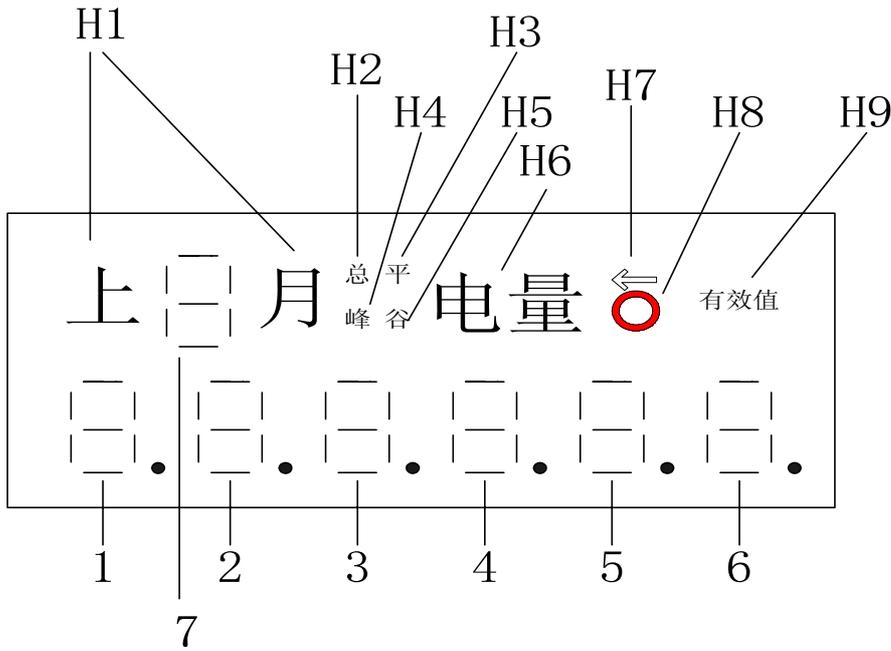
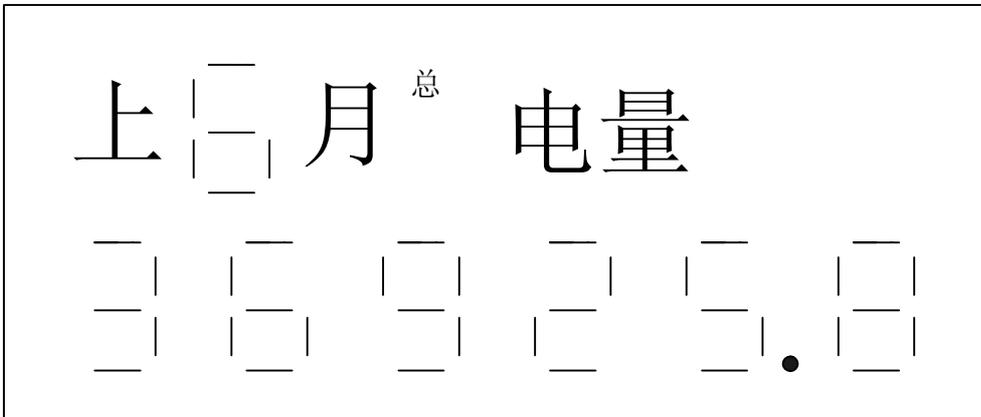


表 : 玻璃真值表

	S0	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15
COM0	7C	7E	H5	H9	1E	1P	2E	2P	3E	3P	4E	4P	5E	5P	6E	6P
COM1	7B	7G	H3	H8	1G	1C	2G	2C	3G	3C	4G	4C	5G	5C	6G	6C
COM2	7A	7F	H2	H7	1F	1B	2F	2B	3F	3B	4F	4B	5F	5B	6F	6B
COM3	H1	7D	H4	H6	1D	1A	2D	2A	3D	3A	4D	4A	5D	5A	6D	6A

若我们需要显示下面的画面，



对应的真值表应该为:

	S0	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15
COM0	1	1	0	0	0	0	1	0	0	0	1	0	0	1	1	0
COM1	0	1	0	0	1	1	1	1	1	1	1	0	1	1	1	1
COM2	1	1	1	0	0	1	1	0	1	1	0	1	1	0	1	1
COM3	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1

所以，针对 LCD 数据寄存器，我们需要填写的数据如下:

寄存器	8810H	8811H	8812H	8813H	8814H	8815H	8816H	8817H
数据	FBH	12H	75H	5FH	77H	3DH	D7H	7FH

5.3 寄存器

•8824H --- LCD 控制寄存器

Write Only

	----	----	----	----	DIS	LCDC1	LCDC0	LCDN
Reset	X	X	X	X	0	0	0	0

LCDN = 0: 禁止 LCD Driver ;

=1: 允许 LCD Driver。

LCDC1 LCDC0 选择 LCD 驱动时钟频率。

0	0	512Hz
0	1	256Hz
1	0	128Hz
1	1	64Hz

DIS: =0: LCD驱动电压VLCD=5V; =1: VLCD=3V。

对应的电压:

DIS=0: VLCD=5V V1=3.3V V2=1.6V

DIS=1: VLCD=3V V1=2V V2=1V

注: 无论用3V或5V的LCD, 在IdleMode都应该写 DIS=0。

•884AH --- 显示方式寄存器

Read/Write

	LCD1	LCD0	SPC1	SPC0	D/NC	DisMode2	DisMode1	DisMode0
Reset	0	0	0	0	0	0	0	0

LCD1 LCD0 DisMode2 DisMode1 DisMode0

管脚复用方式

0	0	0	0	0	4×24LCD, Pin78~85、86~94 用作 GPIO PortD、E
0	1	0	0	0	4×32LCD, Pin86~94 用作 GPIO PortE
1	0	0	0	0	4×32LCD, Pin78~85 用作 GPIO PortD
1	1	0	0	0	4×40LCD
0	0	0	0	1	LED 方式, Pin52~67 用作位、段扫描输出, 其他作为 GPIO PortC/D/E/F
0	0	0	1	0	计度器两相两拍输出方式, Pin95~98 用作 F1~F4, Pin52~94 用作 GPIO PortA/B/C/D/E
0	0	0	1	1	计度器两相四拍输出方式, Pin86~89、91~94 用作 FA1、FA2、FA3、FA4、FB1、FB2、FB3、FB4, 其他作 PortA/B/C/D/F
0	0	1	X	X	I/O 模式, Pin52~98 全部用作 GPIO PortA/B/C/D/E/F

注意: 只有在 (00010) 和 (00011) 两种情况下, D/NC 才有效。

6 LED 部分

这部分主要介绍 ATT7023 的 LED 显示功能。

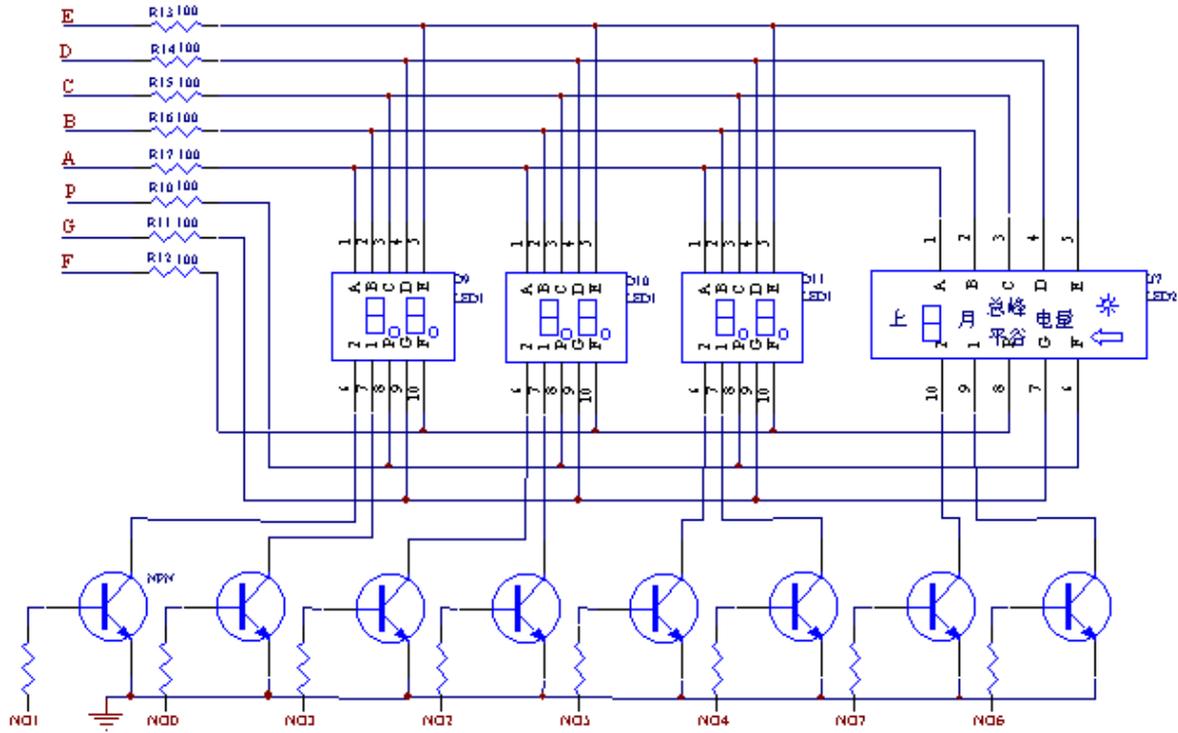
内 容

页码

6.1 功能简述.....	30
6.2 编程指南.....	31

6.1 功能简述

在 ATT7023 中提供 8 位 8 段 LED 译码扫描功能，不含驱动，需要外部加三极管驱动电路。扫描频率为 512HZ。输出由 8 根数据段信号线和 8 根位控制线组成，分别是 A,B,C,D,E,F,G,P 和位控制线 NO0-NO7，与 LCD 显示复用 SEG0-SEG15 输出。内部由不同电路驱动，在寄存器 884A 中的控制位选择电路。待显示的数据由显示数据寄存器给出。LED 显示可以使用共阴极接法。下面是接线图的一例：



6.2 编程指南

使用之前，必须使用 884AH 寄存器选择显示方式为 LED 驱动。

对每一个 8 段 LED，只需要将需要显示的数据写入对应的显示寄存器即可。另外，ATT7023 还支持字符“AbCdEF”显示。对图标部分，我们支持 8 个图标，寄存器 882FH 中的每一个位分别控制一个图标。

例如，我们需要显示“36925.8 上月总电量”，我们需要的操作如下：

- 1、向 884AH 寄存器写入 01H；
- 2、向 8828H, 8829H, 882AH, 882BH, 882DH, 882EH 寄存器分别写入：03, 06, 09, 02, 05, 08。
- 3、向 882CH 寄存器写入 85（将 Bit7 置位，显示小数点。）
- 4、“上月”对应小数点，“总”对应 A 段，“电量”对应 E 段，则向 882FH 寄存器填写 C4。

• 8828H-882FH --- LED 显示数据寄存器 **Write Only**

8828H	第 1 个 LED 显示数据缓冲（编号说明详见 5.5）							
8829H	第 2 个 LED 显示数据缓冲							
882AH	第 3 个 LED 显示数据缓冲							
882BH	第 4 个 LED 显示数据缓冲							
882CH	第 5 个 LED 显示数据缓冲							
882DH	第 6 个 LED 显示数据缓冲							
882EH	第 7 个 LED 显示数据缓冲							
882FH	P	G	F	E	D	C	B	A

7 段 LED 显示	寄存器数值							
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0		X	X	X	0	0	0	0
1		X	X	X	0	0	0	1
2		X	X	X	0	0	1	0
3		X	X	X	0	0	1	1
4		X	X	X	0	1	0	0
5		X	X	X	0	1	0	1
6		X	X	X	0	1	1	0
7		X	X	X	0	1	1	1
8		X	X	X	1	0	0	0
9		X	X	X	1	0	0	1
A		X	X	X	1	0	1	0
B		X	X	X	1	0	1	1
C		X	X	X	1	1	0	0
D		X	X	X	1	1	0	1
E		X	X	X	1	1	1	0
F		X	X	X	1	1	1	1

这些寄存器(除 882FH)采用 0~F 译码，LED 数码管 P 段(小数点)的显示由 Bit7 决定。寄存器 882FH 用于不作译码，直接控制显示符号 (Bit7~0: 1 to display, 0 to hide)，可以支持显示 8 种符号。

7 UART 功能

这部分主要介绍 ATT7023 的 UART1 通讯功能。

内 容

页码

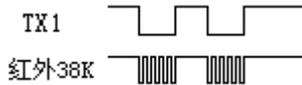
7.1 功能简述.....	33
7.2 编程指南.....	33
7.2 寄存器.....	33

7.1 功能简述

ATT7023 内部集成有两个功能很强的全双工串行通讯口，属 UART 方式，其中 UART0 用 8051 内部的 UART0, UART1 在 8051 的外部。

UART1 中 FIFO（先进先出）是 UART 的数据缓冲器，可以存放 8 个数据。UART 的协议包括一个起始位，5~8bits 的数据位，1 个奇偶校验位和 1~2 个停止位。在 TX/RX 上最先输出最低有效位。

ATT7023 内部有一个由 6M 时钟分频得到的 38K 红外调制振荡波，可以配合 UART1 的 TX1 管脚输出，由 883AH 寄存器的 bit6 (SharpIRDis) 控制。若 SharpIRDis = 1，则 TX1 管脚输出的是原始的 UART1 信号，若 SharpIRDis = 0，则 TX1 管脚输出的是经过 38K 振荡波调制过的 UART1 信号。调制信号与调制前信号的对应示意图如下：



7.2 编程指南

为了简化串行口的使用，UART1 里面，起始位必须为 0，停止位必须为 1。在与其他外设进行通讯之前，必须把波特率、数据位数、有无奇偶校验等协议设置成一致。

需要注意的是：发送数据时，发送两个停止位，接收时，能检测一个或多个停止位。

使用 UART1 时，所用到的资源为：

- 1、中断控制寄存器，中断状态寄存器：8844H，8846H；
- 2、UART1 波特率选择寄存器：8838H，8839H；
- 3、UART1 控制寄存器：883AH；数据寄存器 883BH；状态寄存器 883CH。

UART1 的工作步骤：

- 1、程序设定好 8051 的中断允许寄存器 IE，以及 TCON 的相关位；
- 2、程序向波特率选择寄存器 8838H 和 8839H 写入相应数据，设定好波特率。
- 3、程序向 UART1 控制寄存器写入相应数据，设定好的数据位数，校验方式和是否为红外通讯，并设定好发送、接收中断允许，以及中断模式控制（883CH bit6）。
- 4、程序设定好 ATT7023 的中断控制寄存器 8846H 中相应控制位 IC4，允许 UART1 中断。
- 5、如果是发送数据，则向 UART1 FIFO 数据寄存器 883BH 写入需要发送的数据。根据 UART1 FIFO 状态寄存器 883CH 的内容，可以判断一组数据发送是否完成。
- 6、如果是接收数据，则直接根据 UART1 FIFO 状态寄存器 883CH 的内容，判断是否有数据接收成功。
- 7、在接收中断服务处理程序中，应该判断 883CH 的 bit3，当 RX FIFO 为空时，才退出中断服务程序。
- 8、在发送时，向 883BH TX FIFO 中写数据，应该判断 883CH 的 bit2，当 TX FIFO 为满时，必须先等到 883CH 的 bit2 变为 0，才能继续往 883BH 中写数据。
- 9、在发送数据之前，要将 883AH 的 TIE (bit7) 置 1，在数据发送完之后，要先将 883AH 的 TIE 清零，再清 883CH 中的 TX 中断标志。
- 10、在发送数据的过程中，当把需要发送的最后一个字节的数据写入 883BH 之后，要等待一段时间或者程序查询到发送完成的标志位之后才能去清 883AH 和 883CH 中的相应位。假如用 2400bps 的波特率，则发送 8 字节（64 位）的数据需要等待的时间为： $64/2400 = 27\text{ms}$ 。

★ 为保证传输数据的正确性，波特率的值应该小于 3600bps。

7.3 寄存器

请参考附录 1：所有寄存器列表中 8838H~883CH 寄存器。

8 SPI 功能

这部分主要介绍 ATT7023 的 SPI 串行通讯功能。

内 容

页码

8.1 功能简述.....	35
8.2 编程指南.....	35
8.2 寄存器.....	36

8.3 寄存器

●**8830H** --- **SPI Data Buffer 低字节** **Read/Write**

	SD7	SD6	SD5	SD4	SD3	SD2	SD1	SD0
Reset	X	X	X	X	X	X	X	X

●**8831H** --- **SPI Data Buffer 中字节** **Read/Write**

	SD15	SD14	SD13	SD12	SD11	SD10	SD9	SD8
Reset	X	X	X	X	X	X	X	X

●**8832H** --- **SPI Data Buffer 高字节** **Read/Write**

	SD23	SD22	SD21	SD20	SD19	SD18	SD17	SD16
Reset	X	X	X	X	X	X	X	X

●**8833H** --- **SPI 命令寄存器** **Read Only**

	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
Reset	0	0	0	0	0	0	0	0

COM7: (read or write) =0, SPI 向外部送出数据
 =1, SPI 接收外部读数据

COM6~COM0: 命令格式由软件定义。

9 电源监控

这部分主要介绍 ATT7023 的电源监控以及在电源电压变化时各种工作模式的转换情况，同时对 Standby 和 Idle 两种模式作详细描述。

内 容

页码

9.1 功能简述.....	38
9.2 Standby Mode 和 Idle Mode	38
9.2.1 ATT7023 在停电状态下的工作情况	38
9.2.2 Standby Mode	38
9.2.3 Idle Mode	38
9.2.4 Idle Mode 编程指南.....	39
9.3 工作模式控制.....	40
9.4 电源供电系统.....	40

9.1 功能简述

ATT7023 片内包含一个电源监控电路，连续对模拟电源（AVCC）和数字电源（VCC）进行监控。当数字电源电压低于 4.1V 时，ATT7023 内部产生一个中断信号，作为 8051 的外部中断 0 输入，产生电源掉电中断，8051 在中断程序中可以完成保存数据的工作。

在电源上升的过程中，当数字电源电压高于 4.1V 时，ATT7023 内部也会产生一个中断信号，与电源下降时低于 4.1V 所产生的中断属于同一个中断。

电源电压低于 4.1V 时，计量部分停止计量，不管各通道是否有输入信号，都不累积电能读数。

停电后 ATT7023 进入 Standby Mode，此时由后备电池供电；通过外部信号（SBSW_引脚）可以将 8051 激活，使 ATT7023 进入 Idle Mode，此时 8051 也由后备电池供电。

9.2 Standby Mode 和 Idle Mode

9.2.1 ATT7023 在停电状态下的工作情况

ATT7023 在停电状态下有两种工作模式：Standby Mode 和 Idle Mode，停电后进入 Standby Mode，如果有外部请求（停电抄表）发生，则进入 Idle Mode。

ATT7023 在停电后自动进入待机模式（Standby Mode），此时其工作特性如下：

- ① 256 byte 的 SRAM（Standby SRAM）不掉电
- ② RTC 模块不掉电，最低工作电压是 3.0V，典型值：3.5V
- ③ 其它电路的电源全部被关掉
- ④ 8051 可以被外部信号（SBSW_引脚）激活
- ⑤ 整体电路的耗电小于 5uA

在 Standby Mode 下 8051 被激活后，ATT7023 进入 Idle Mode，此时其工作特性如下：

- ① 8051 的 RAM、ROM 恢复供电，8051 被 reset
- ② 在外接 LCD 的时候，LCD 能够显示
- ③ 计量模块的电源被关掉
- ④ SPI、UART1 的电源也被关掉
- ⑤ 能够回到 Standby Mode（由软件写系统状态寄存器控制）
- ⑥ 整体电路的耗电小于 40uA

9.2.2 Standby Mode

当数字电源电压低于 4.1V 时，ATT7023 内部产生一个中断信号，作为 8051 的外部中断 0 输入，产生电源掉电中断。当数字电源电压低于 3.0V 时，ATT7023 内部产生一个 reset 信号，使除了 RTC、Standby SRAM、SBSW_Pin 之外的所有电路复位，ATT7023 进入 Standby Mode，此时 RTC、Standby SRAM、SBSW_Pin 由 VBAT 引脚供电。

留出一个 Pin（SBSW_）作为中断时（停电抄表）唤醒内部电路的接口，如果不用停电抄表功能，此 Pin 可以固定接高。在 Standby Mode，当此 Pin 接收到一个脉冲时，在芯片内部 VDD 和 VBAT 之间有一个开关将被打开，VDD 将由 VBAT 供电（VCC 和 AVCC 仍然没有电），因此就恢复 8051、RAM、ROM、LCD Driver、的供电，进入 Idle Mode；在 ATT7023 正常工作和 Idle Mode 时，此 Pin 接收的脉冲无效。

9.2.3 Idle Mode

- ① 此时 8051 工作在低频晶振频率（32768Hz）下；
- ② 此时 SBSW_Pin 上的脉冲无效；
- ③ ATT7023 在 Idle Mode 的工作时间不应该超过一个固定的时间（软件控制），即此时间后 ATT7023 能够通过用软件写 8808H 寄存器的 IDLEM = 0 而回到 Standby Mode；
- ④ 在正常工作时如果写 IDLEM = 1 也可以进入 Idle Mode，这样可以有下面两个作用：

- a. 停电时进入电源掉电中断后，如果写 IDLEM = 1，可以马上进入 Idle Mode，从而控制屏幕的轮显等功能，而不需要通过按键再进入 Idle Mode。
- b. 调试 Idle Mode 程序时，可以直接用仿真器模拟 Idle Mode 的运行条件(8051 工作在低频状态)。
- ⑤ 在 Idle Mode 时来电，ATT7023 不 reset，至 4.1V 会产生 IRQ。通过软件可以在上电中断服务程序中复位 8051。

9.2.4 Idle Mode 编程指南

综上所述，有两种方法可以使 ATT7023 进入 Idle 工作模式：

- ① 在 Standby Mode 时，给 SBSW_ 一个低脉冲；
在应用时给 SBSW_ 接一个按键，可以实现停电抄表。在 Standby Mode 按下时，就会进入 Idle Mode。由于在 Normal Mode 和 Idle Mode，SBSW_ 无效，因此此按键同时还可以复用为其它的功能（例如轮显）。
- ② 在 Normal Mode 时，写 IDLEM = 1。
这种应用情况是完全通过软件来控制进入 Idle Mode 的，与 SBSW_ 引脚无关。其实现停电抄表的方法是：在停电后的掉电中断服务程序中，写 IDLEM = 1，使 ATT7023 进入 Idle Mode，由软件控制点亮 LCD 一段时间（可以用定时器或者按键使 LCD 轮显），然后退出 Idle Mode。

Idle Mode 工作步骤：

- 1、在程序开始进入 Normal 程序主循环之前，查询 8808H 寄存器 bit0 和 bit4 的状态，如果 bit0 = 1 and bit4 = 0，则进入 Idle 程序主循环，否则进入 Normal 程序主循环。
- 2、在进入 Idle 程序之后，应将 LCD 的工作电压设为 5V 模式，即给 8824H 的 bit3 写 0。
- 3、在 Idle mode 下，请注意 8051 内部的两个定时器 T0 和 T1 的时钟源改为 32768Hz，8051 外部的两个定时器 T2 和 T3 的时钟源除了 8842H 的 [TSS31 TSS30] = 11 和 [TSS21 TSS20] = 11 外，其它情况不变。
- 4、在 Idle 程序主循环中，为了程序的安全性考虑，应该有判断 8808H 的 bit4 的语句，如果 bit4 = 1，则跳回至程序复位入口处。
- 5、由于上电时的电源波动，在程序开始时判断 8808H 寄存器的状态可能会不可靠，因此为程序的安全性考虑，在 Normal 程序的主循环中，应该有判断 8808H 的 bit4 的语句，如果 bit4 = 0，则跳回至程序复位入口处
- 6、在 Idle 程序中，当退出 Idle Mode 的定时时间到时，写 8808H 寄存器的 bit0 = 0 即可退出 Idle Mode。之后建议 8051 在一个空操作循环中判断 8808H 寄存器 bit0 的状态，不需要作其它操作。如下面程序所示：

```
mov DPTR, #8808H
movx A, @DPTR
anl A, #0FEH
movx @DPTR, A ; 给 8808H 寄存器 bit0 写 0 以退出 Idle Mode
```

LIdle4:

```
nop
nop
nop
nop
mov DPTR, #8808H ; 判断 8808H 寄存器 bit0 的状态，为 0 则继续空操作循环
movx A, @DPTR
jnb ACC.0, LIdle5
jmp Lmain ; 为 1 则跳回至程序复位入口处
```

LIdle5:

```

nop
nop
nop
nop
sjmp LIdle4

```

- 7、 由于在 Idle Mode， ATT7023 内部的 VCC 没有电压， VDD 的电压是由 VBAT 而来， 做成整表后的耗电典型值为 35uA， 如果用 I/O 口作为 Idle Mode 下轮显等功能的按键， 请注意 I/O 口的上拉电阻应该在 1M（接到 VDD） 以上， 以达到省电目的。

9.3 工作模式控制

ATT7023 的 IDLE Mode 也可以被禁止不用， 被禁止后 SBSW_上的按键无效， 用软件写 IDLEM = 1 也不会进入 Idle Mode。

从工作方式来讲， ATT7023 又可以分为正常工作模式和开发模式， 开发模式是用 ATT7023 的开发板进行软件开发时， 所用的一种工作方式， 此时 ATT7023 内部的 8051 被映射到 ATT7023 的外部。 正常工作模式即 ATT7023 内部的 8051 被正常使用。

无论正常工作模式和开发模式， 都可以根据 TEST、MODE1 和 MODE0 这三个 Pin 来选择是否禁止 IDLE Mode。

TEST	MODE1	MODE0	ATT7023 工作模式
1	0	0	开发模式， 不支持 IDLE Mode
1	0	1	正常工作模式， 不支持 IDLE Mode
1	1	0	开发模式， 支持 IDLE Mode
1	1	1	正常工作模式， 支持 IDLE Mode

TEST、MODE1 和 MODE0 这三个 Pin 由用户根据自己的应用情况来选择接高电平还是接低电平， 只需在 PCB 板上通过 10K 的电阻接到 VDD 或者 GND 即可。

9.4 电源供电系统

ATT7023 有五组电源： VCC、VDD、AVCC、VBAT、VPP， 它们分别对 ATT7023 内部不同的模块供电：

	VCC	VDD	AVCC	VBAT	VPP
供电模块	计量单元数字电路 高频晶振 UART1 串行接口 SPI 接口 4.1V 电压监测单元 LED 驱动单元 通用 I/O 口： E 和 F	8051 外部 2K 的 SRAM 32K 的 OTP ROM LCD 单元 定时器单元 看门狗单元 3.0V 电压监测单元 通用 I/O 口： A、B、C、D、G、H 外部中断： EIRQ0 和 EIRQ1	计量单元 模拟电路	时钟单元 低频晶振 SBSW_引脚 掉电 SRAM	OTP 编程 电压输入， 应用时接 VDD

10 实时时钟和定时器控制部分

这部分主要介绍 ATT7023 的与实时时钟有关的功能,包括 RTC、Timeout 脉冲输出、以及定时器(Timer)部分。

内 容

页码

10.1	功能简述.....	42
10.2	RTC 部分编程指南.....	42
10.3	定时器部分编程指南.....	42
10.4	寄存器.....	42

10.1 功能简述

时钟控制部分，包括 ATT7023 提供的低功耗实时时钟和时钟调校电路、两个 8 位计时器、可选脉冲输出、实时时钟中断。

低功耗实时时钟单元（Real Time Clock）外接 32768Hz 晶振，能自动判断闰年及每个月的天数，内含 256 字节的低功耗 SRAM（8900H ~ 89FFH），在停电时由电池供电。

利用实时时钟，还可以设定中断，间隔可选（1 秒/1 分/1 小时/1 天）。

ATT7023 还提供实时时钟调校功能，使得实时时钟的精度能够保持在 3.05ppm。

为了方便校准时间，ATT7023 还提供了一个可选频率脉冲输出（Timerout），参见寄存器 8849H。

ATT7023 除了可以使用 8051 的 T0, T1 两个计时器外，另外提供两个 8 位定时器，可以用作程序控制，我们称之为 Timer2 和 Timer3。所有的这四个计时器的时钟源是可选的，参见计时器控制寄存器 8842H。

10.2 RTC 部分编程指南

针对实时时钟，我们仅仅需要在第一次上电时，对相关的寄存器写入正确的时间数据就可以了。如果有必要，以后可以通过读 EEPROM，UART，SPI 等方式进行重新校正。

对 ATT7023 的时钟寄存器（8800H - 8805H）进行操作时，应该注意的是：ATT7023 的时钟寄存器都是以 0 为起始点开始计时，而在实际中年、小时、分、秒是以 0 为起始点，月和日是以 1 为起始点，因此对月和日寄存器进行操作时，写入的数值需要减 1，读出的数值需要加 1。例如要写 3 月 2 日到月和日寄存器中，应该给月寄存器写 02H，给日寄存器写 01H。

读时间寄存器时，要注意屏蔽不用的位，比如读秒寄存器，应该和 0X3F 相与。

10.3 定时器部分编程指南

ATT7023 总共有 4 个定时器，包括 8051 中的两个分别为 Timer0 和 Timer1，另外内建两个 8 位的定时器分别为 Timer2 和 Timer3，这里主要介绍 Timer2 和 Timer3 的操作：

- (1) 向 8842H 写入正确的值；
- (2) 清零 INTO 中断状态寄存器 8844H 的 IF6/IF7；
- (3) INTO 中断允许并设置为电平触发方式（SETB EX0, CLR IT0, SETB EA）；
- (4) 向 8840H 或 8841H 中装入初值；
- (5) 向 INTO 中断允许寄存器 8846H 的 IC6/IC7 写 1，允许 Timer2/Timer3 中断。

10.4 寄存器

请参考附录 1：所有寄存器列表。

RTC 寄存器：8800H~8806H

RTC 中断时间间隔选择寄存器：8809H

Timer 寄存器：8840H~8842H

Timeout 脉冲输出控制寄存器：8849H

11 通用 I/O 口

这部分主要介绍 ATT7023 的通用 I/O 口的功能及用法。

内 容

页码

11.1 功能描述.....	44
11.2 寄存器.....	44

11.1 功能描述

ATT7023 包括 A、B、C、D、E、F、G、H 总共 8 组通用 I/O 口，其中 A、B、C、D、E、G 分别有 8 根 I/O 口线，F、H 分别有四根 I/O 口线。A、B、C、D、E、F 的管脚和 LCD 的 COM 和 SEGMENT 复用，因此当使用 LCD/LED/计度器方式输出时，这几组 I/O 口将有一些不能使用，而不论使用何种输出方式，G、H 口始终有效。详细的使用情况见 884AH 寄存器的定义。

每组 I/O 口都有两个寄存器：控制寄存器和数据寄存器，从下面的寄存器列表可以看到所有的 I/O 默认都是输出低电平。在编程时，需要先设置 884AH 寄存器以选择使用的 I/O 口，然后对控制寄存器和数据寄存器进行操作即可。

ATT7023 的通用 I/O 口驱动电流的能力 $\geq 5\text{mA}$ 。

11.2 寄存器

884AH 寄存器的 LCD1、LCD0、DisMode2、DisMode1、DisMode0 确定了 I/O 口的使用情况。如下表所示：

MODE	L1	L0	D2	D1	D0	A	B	C	D	E	F	G	H
LCD	0	0	0			Seg	Seg	Seg	IO	IO	Com	IO	
	0	1				Seg	Seg	Seg	Seg	IO	Com		
	1	0				Seg	Seg	Seg	IO	Seg	Com		
	1	1				Seg	Seg	Seg	Seg	Seg	Com		
LED	0		0	0	1	LED 信号		IO	IO	IO	IO	IO	
两相两拍计度器			0	1	0	IO	IO	IO	IO	IO	计度器信号		
两相四拍计度器			0	1	1	IO	IO	IO	IO	计度器信号	IO		
通用 I/O 口			1	X	X	IO	IO	IO	IO	IO	IO		

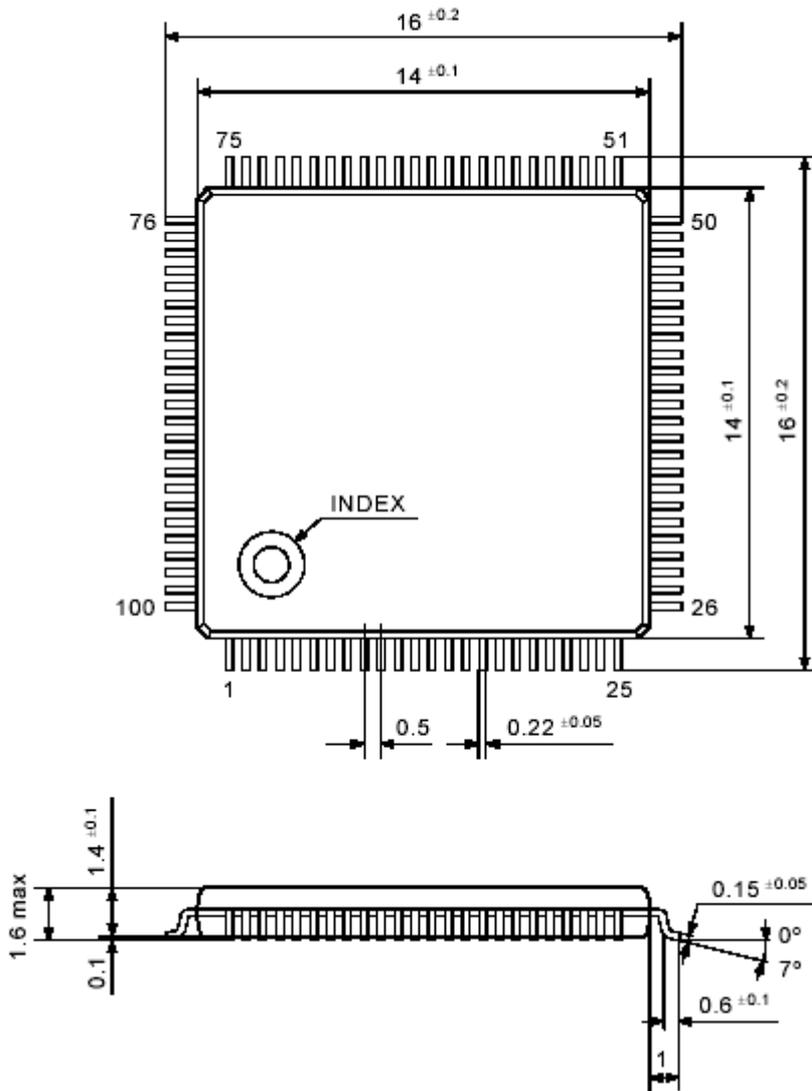
注：其中 L1、L0、D2、D1、D0 分别表示 884AH 寄存器的位：LCD1、LCD0、DisMode2、DisMode1、DisMode0。

其它寄存器的详细情况请参考附录 1：所有寄存器列表中 8850H~885FH 寄存器。

12 芯片封装

ATT7023 采用 100PIN LQFP 封装。

Unit : mm



13 开发工具

ATT7023 需要由用户自行开发软件，我们提供相配套的软件开发和写码的工具，包括：一块软件开发板，一块编程器，各自有独立的说明文档，可以从我公司网站上下载。

内 容

页码

13.1	软件开发.....	47
13.2	写码工具.....	47
13.3	FirmWare 保密设置	47
13.4	OTP ROM 出厂设置	47

13.1 软件开发

我们提供一套开发平台，可以直接放在任何一款现成的 8051 仿真器上。该平台提供 ATT7023 除 8051 以及 32K 的 OTP ROM 外其它所有的资源。由于市场上 8051 的仿真器已经非常普遍，因此我们不提供 8051 的仿真器（需要用户自己准备）。

软件开发完成之后，将 code 通过我们提供的编程器烧入 ATT7023 的 OTP ROM 即可。

13.2 写码工具

我们提供一套烧录装置，可以直接对 ATT7023 的芯片进行程序烧录。

13.3 FirmWare 保密设置

为防止用户 FirmWare 被非法 copy，设一 8 Bit 密码寄存器（也是 OTP 的，放在程序 ROM 空间，占用地址 7FFFH），Default 状态下（全为“0”），可读 EPROM 里的内容。当程序写入 OTP ROM，烧码器校验无误之后，再进行“写保密位”操作，写一非全“0”数到密码寄存器，之后 ROM 将被永远锁死不可读。

13.4 OTP ROM 出厂设置

芯片出厂时 OTP ROM 的两个字节 7FFDH - 7FFEh 已经被占用，在编程时应该注意，程序不能存放到这两个字节。

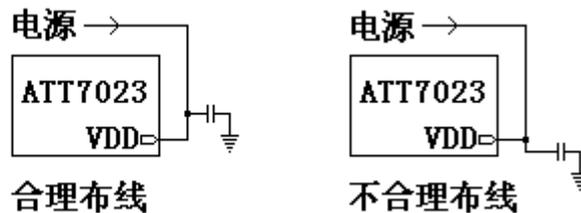
14 应用线路图及 PCB 注意事项

在您使用 ATT7023 设计多功能电能表时，请参考下面的应用线路图，您的设计可能和下面的例子有所不同，但是基本的线路和参数应该和我们推荐的应用线路图保持一致：

- (1) ATT7023 有五组电源：VCC、VDD、AVCC、VBAT、VPP。在设计电源系统时应该注意这些电源要区分开来，正常工作时各组电源电压都是 5V，两组电源之间的压差不能超过 0.4V。在 Idle Mode 时，VBAT 和 VDD 有 3V 电压，而 VCC 和 AVCC 是没有电的，为了保证系统的省电，在应用原理图上，要注意不能有从 VBAT 和 VDD 漏电到 VCC 和 AVCC 的情况。VPP 是 OTP 的编程电压输入，应用时直接接到 VDD 即可。
- (2) ATT7023 可以通过 TEST、MODE1、MODE0 三个引脚来选择是否使用 Idle Mode，如果使用 Idle Mode，在应用原理图上，要严格注意电源的接法，在 ATT7023 内部各部分电源所供电的模块请参考 9.4 节。
- (3) 低频晶振的外接电阻应该用 10M Ω ，补偿电容 50pF。
- (4) 在 Idle 模式下需要使用的按键的上拉电阻应该上拉到 VDD，且阻值在 1 M Ω 以上。
- (5) 7805 后面所接的大电容推荐值：2200 μ F。

在 PCB 布线时，应该遵循以下几点：

- (1) ATT7023 内部有两组地：GND 和 AGND，分别为数字和模拟电源参考点。在原理图上可以直接连接，在 PCB 板上应该将它们就近接地，形成大面积多点接地，接地点不要用电感、电阻和磁珠等，这样可以保证良好的抗电脉冲性能。
- (2) 晶振的信号线附近布大面积地，不要有其他信号线从中穿过。
- (3) 电流输入通道的布局要对称，以保证计量线性度。
- (4) 火线和零线不能从芯片上方走过，也不能靠近芯片。
- (5) EEPROM 尽量靠近 ATT7023，控制 EEPROM 的 IO 口线不能走太长。
- (6) 在 7805 前面的高压部分的下面不应该大面积铺地，以防高压信号对地线的干扰。
- (7) 电源前面的滤波电容应该走在 IC 管脚之前，以免达不到滤波效果。如下图所示：



附录 1: 所有寄存器列表

1. 功能寄存器

•8800H --- **RTC Second Counter** **Read/Write**

	--	--	Sc5	Sc4	Sc3	Sc2	Sc1	Sc0
--	----	----	-----	-----	-----	-----	-----	-----

实时时钟的秒计数器。

•8801H --- **RTC Minute Counter** **Read/Write**

	--	--	Mn5	Mn4	Mn3	Mn2	Mn1	Mn0
--	----	----	-----	-----	-----	-----	-----	-----

实时时钟的分钟计数器。

•8802H --- **RTC Hour Counter** **Read/Write**

	--	--	--	Hr4	Hr3	Hr2	Hr1	Hr0
--	----	----	----	-----	-----	-----	-----	-----

实时时钟的小时计数器。

•8803H --- **RTC Date Counter** **Read/Write**

	--	--	--	Da4	Da3	Da2	Da1	Da0
--	----	----	----	-----	-----	-----	-----	-----

实时时钟的日计数器。

•8804H --- **RTC Month Counter** **Read/Write**

	--	--	--	--	Mon3	Mon2	Mon1	Mon0
--	----	----	----	----	------	------	------	------

实时时钟的月计数器。

•8805H --- **RTC Year Counter** **Read/Write**

	--	Year6	Year5	Year4	Year3	Year2	Year1	Year0
--	----	-------	-------	-------	-------	-------	-------	-------

实时时钟的年计数器。(2000~2099)

•8806H --- 实时调校寄存器 (可以放入 EEPROM 中) **Read/Write**

	D7	D6	D5	D4	D3	D2	D1	D0
Reset	0	0	0	0	0	0	0	0

IC 内每 10 秒加减 N 个低频晶振 clock, N 是 D6-D0 的补码, D7 为符号位, D7=0 为减 clock, D7=1 为增加 clock。最小解析度为 $1/32768 \times 10 = 1/0.32768\text{ppm}$ 。例如: 此寄存器值为 02h, 则表示每 10 秒减 2 个低频晶振 clock, 即每 10 秒时钟减慢 $2/32768 = 61\mu\text{s}$ 。

调校方法: 客户根据 TIMEOUT 管脚的输出 (此时选为 32768Hz 输出) 确定 32.768K 时钟的误差, 得到写入此寄存器的增益值, 通过红外通信、485 通信或者 EEPROM 等方法将该增益值写入此寄存器, 然后硬件作出相应增减 clock 动作。

•8807H --- 低频晶振控制寄存器 **Read/Write**

	--	--	0	0	--	--	0	0
--	----	----	---	---	----	----	---	---

此寄存器控制 32768Hz 低频晶振的强弱, 写入的数值越大, 功耗就越小, 但是低频晶振的抗干扰能力会稍弱; 反之, 写入的数值越小, 功耗就越大, 但是低频晶振的抗干扰能力会增强。其中 bit0, bit1, bit4, bit5 固定为 0, 这样写入的值只能在 00H~CCH。

•8808H --- 系统状态寄存器

	--	--	--	IRQS	RESS	--	--	IDLEM
--	----	----	----	------	------	----	----	-------

IDLEM: =0, 表示 ATT7023 目前没有工作于 Idle Mode

=1, 表示 ATT7023 目前正工作于 Idle Mode, **Read/Write**

注意: 写 IDLEM = 1 进入 Idle Mode, 写 IDLEM = 0 退出 Idle Mode。如果在 Normal Mode 写 1 进入 Idle Mode, 则只切换 8051 的晶振, 不会产生系统 reset。

RESS: =0, 表示数字电源电压低于 3.0V

=1, 表示数字电源电压高于 3.0V。Read Only

IRQS: =0, 表示数字电源电压低于 4.1V

=1, 表示数字电源电压高于 4.1V。Read Only

•8809H --- RTC 中断时间间隔选择寄存器 **Read/Write**

	----	----	----	----	----	----	RTC1	RTC0
Reset	X	X	X	X	X	X	0	0

RTC 中断时间间隔选择(仅仅影响产生 IRQ13 的时间间隔, 不影响 RTC 的计数)

RTC1, 0 = 00, 1 second;

= 01, 1 minute;

= 10, 1 hour;

= 11, 1 day;

•8810H~8823H --- LCD 寄存器 **Read/Write**

Data		Bit7	Bit6	Bit5	Bit4		Bit3	Bit2	Bit1	Bit0
REG		COM0	COM1	COM2	COM3		COM0	COM1	COM2	COM3
8810H	SEG1					SEG0				
8811H	SEG3					SEG2				
8812H	SEG5					SEG4				
8813H	SEG7					SEG6				
8814H	SEG9					SEG8				
8815H	SEG11					SEG10				
8816H	SEG13					SEG12				
8817H	SEG15					SEG14				
8818H	SEG17					SEG16				
8819H	SEG19					SEG18				
881AH	SEG21					SEG20				
881BH	SEG23					SEG22				
881CH	SEG25					SEG24				
881DH	SEG27					SEG26				
881EH	SEG29					SEG28				
881FH	SEG31					SEG30				
8820H	SEG33					SEG32				
8821H	SEG35					SEG34				
8822H	SEG37					SEG36				
8823H	SEG39					SEG38				

20 bytes for 40 Segment × 4 Common. Bit = 1 to display, 0 to hide.

•8824H --- LCD 控制寄存器

Read/Write

	----	----	----	----	DIS	LCDC1	LCDC0	LCDN
Reset	X	X	X	X	0	0	0	0

LCDN = 0: LCD Driver disabled; =1: enable LCD Driver.

LCDC1 LCDC0 选择 LCD Source Clock Frequency

0	0	32768/2 ⁶
0	1	32768/2 ⁷
1	0	32768/2 ⁸
1	1	32768/2 ⁹

frame frequency对应如下:

LCD Clock Frequency Display Duty Ratio	32768/2 ⁹ (64Hz)	32768/2 ⁸ (128Hz)	32768/2 ⁷ (256Hz)	32768/2 ⁶ (512Hz)
	1/4	16Hz	32Hz	64Hz

BIT3 DIS: =0: VLCD=5V; =1: VLCD=3V。

对应的电压:

DIS=0: VLCD=5V	V1=3.3V	V2=1.6V
DIS=1: VLCD=3V	V1=2V	V2=1V

•8828H~882FH --- LED 显示数据寄存器

Read/Write

8828H	第 1 个 LED 显示数据缓冲 (编号说明详见 5.5)
8829H	第 2 个 LED 显示数据缓冲
882AH	第 3 个 LED 显示数据缓冲
882BH	第 4 个 LED 显示数据缓冲
882CH	第 5 个 LED 显示数据缓冲
882DH	第 6 个 LED 显示数据缓冲
882EH	第 7 个 LED 显示数据缓冲
882FH	显示 H1~H9

7 段 LED 显示	寄存器数值							
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0		X	X	X	0	0	0	0
1		X	X	X	0	0	0	1
2		X	X	X	0	0	1	0
3		X	X	X	0	0	1	1
4		X	X	X	0	1	0	0
5		X	X	X	0	1	0	1

6		X	X	X	0	1	1	0
7		X	X	X	0	1	1	1
8		X	X	X	1	0	0	0
9		X	X	X	1	0	0	1
A		X	X	X	1	0	1	0
B		X	X	X	1	0	1	1
C		X	X	X	1	1	0	0
D		X	X	X	1	1	0	1
E		X	X	X	1	1	1	0
F		X	X	X	1	1	1	1

这些寄存器（除 882FH）采用 0~F 译码，LED 数码管 P 段（小数点）的显示由 Bit7 决定。寄存器 882FH 用于不作译码，直接控制显示符号（Bit7~0: 1 to display, 0 to hide），可以支持显示 8 种符号。

●**8830H** --- **SPI Data Buffer 低字节** **Read/Write**

	SD7	SD6	SD5	SD4	SD3	SD2	SD1	SD0
Reset	X	X	X	X	X	X	X	X

●**8831H** --- **SPI Data Buffer 中字节** **Read/Write**

	SD15	SD14	SD13	SD12	SD11	SD10	SD9	SD8
Reset	X	X	X	X	X	X	X	X

●**8832H** --- **SPI Data Buffer 高字节** **Read/Write**

	SD23	SD22	SD21	SD20	SD19	SD18	SD17	SD16
Reset	X	X	X	X	X	X	X	X

●**8833H** --- **SPI 命令寄存器** **Read Only**

	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
Reset	0	0	0	0	0	0	0	0

COM7: (read or write) =0, SPI 向外部写数据

=1, SPI 从外部读数据

COM6~COM0: 命令格式由软件定义。

●**8838H** --- **UART1 波特率选择寄存器低字节** **Read/Write**

8 位	BDCoe7	BDCoe6	BDCoe5	BDCoe4	BDCoe3	BDCoe2	BDCoe1	BDCoe0
Reset	0	0	0	0	0	0	0	0

采用时钟除频电路产生 UART1 波特率，此寄存器定义除频的系数。例如：若波特率采用 1200bps，则除频系数=6000000/(1200×8)=625。

●**8839H** --- **UART1 波特率选择寄存器高字节** **Read/Write**

8 位	BDCoe15	BDCoe14	BDCoe13	BDCoe12	BDCoe11	BDCoe10	BDCoe9	BDCoe8
Reset	0	0	0	0	0	0	0	0

•883AH --- UART1 控制寄存器 **Read/Write**

(8 位)	TIE	RIE	STKP	EPS	PEN	SharpIRDis	WL1	WL0
Reset	0	0	0	0	0	0	0	0

Bit 7: UART1 TX 中断允许控制位, =1, 允许; = 0, 禁止

Bit 6: UART1 RX 中断允许控制位, =1, 允许; = 0, 禁止

Bit 5: STKP, 固定校验位

Bit 4: EPS, 偶校验

Bit 3: PEN, 校验位开启控制

PEN	EPS	STKP	校验模式
0	x	x	没有校验位
1	0	0	奇校验
1	1	0	偶校验
1	0	1	校验位为 1
1	1	1	校验位为 0

Bit 2: SharpIRDis : 控制 TX1 输出是否具有 38KHz 的红外调制。

=0, 经过红外调制; = 1, 没有经过红外调制

Bit 1~0: WL[1:0], 帧传输位数控制

WL1	WL0	帧传输位数
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

STOP 控制: 发送时固定产生两个 STOP 位, 接收时可以接收一个以上的 STOP 位。

•883BH --- UART1 FIFO DATA 寄存器 **Read/Write**

	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
Reset	X	X	X	X	X	X	X	X

写这个地址, 会将数据写到 UART1 TX FIFO, 读这个地址, 将从 UART1 RX FIFO 读取数据。

这是 UART1 模块的数据缓冲区, 其内部总共有 16 个字节的缓冲区, 其中 8 个字节用于发送缓冲(TX FIFO), 8 个字节用于接收缓冲(RX FIFO), 内部采用队列(先进先出)的形式管理。例如对 883BH 连续写 8 次就可以写 8 个字节的内容到发送缓冲区; 假如接收缓冲区有 8 个字节的内容, 则对 883BH 连续读 8 次就可以读出 8 个字节的内容。在实际应用中需要结合 883CH 寄存器使用。

•883CH --- UART1\FIFO 状态寄存器 **Read/Write**

	R_ERR	IRQMD	R_IRQ	T_IRQ	RFIFOE	TFIFOE	RFIFOERR	TFIFOERR
Reset	0	0	0	0	1	0	0	0

Bit 7: UART1 接收错误标志, =0 接收正常; =1 接收出错。只有向该位写 1 可以将该位清零。

Bit 6: UART1 FIFO 模式控制

=0 当 TX FIFO 空或者 RX FIFO 中至少有一个字节数据时发出 IRQ。

=1 当 TX FIFO 半空或者 RX FIFO 半满时发出 IRQ

Bit 5: UART1 RX IRQ 中断标志。=1, 有 RX IRQ 中断未处理。只有向该位写 1 可以将该位清零

=0, 没有 UART1 RX IRQ 未处理。

Bit 4: UART1 TX IRQ 中断标志。=1, 有 TX IRQ 中断未处理, 只有向该位写 1 可以将该位清零

=0, 没有 UART1 TX IRQ 未处理。

Bit 3: UART1 RX FIFO 空标志。只读。为 1 表示 UART1 RX FIFO 已被读空。

Bit 2: UART1 TX FIFO 满标志。只读。为 1 表示 UART1 TX FIFO 已被写满，需要等待至少发出一个字节数据后此位才会被清零。

Bit 1: RX FIFO 错误标志。只有向该位写 1 可以将该位清零并复位 FIFO。

Bit 0: TX FIFO 错误标志。只有向该位写 1 可以将该位清零并复位 FIFO。

•8840H --- 8 bit Timer2 Reload Data Write Only

	STA7	STA6	STA5	STA4	STA3	STA2	STA1	STA0
Reset	0	0	0	0	0	0	0	0

8 位的减一定时/计数器，当减至零时，产生中断，并且先前的数据被自动再装入，计数器重新开始计数。当 Timer2 中断被禁止（寄存器 8846H 的 Bit6=0），Timer2 被关闭。

定时时间的计算： $\Delta t = (\text{Data} + 1) / (\text{Clock Source})$

•8841H --- 8 bit Timer3 Reload Data Write Only

	STB7	STB6	STB5	STB4	STB3	STB2	STB1	STB0
Reset	0	0	0	0	0	0	0	0

8 位的减一定时/计数器，当减至零时，产生中断，并且先前的数据被自动再装入，计数器重新开始计数。当 Timer3 中断被禁止（寄存器 8846H 的 Bit7=0），Timer3 被关闭。

定时时间的计算： $\Delta t = (\text{Data} + 1) / (\text{Clock Source})$

•8842H --- 计时器控制寄存器 Write Only

	TSS31	TSS30	TSS21	TSS20	ITS11	ITS10	ITS01	ITS00
Reset	0	0	0	0	0	0	0	0

TSS31 TSS30 Timer3 Clock source

0	0	4Hz
0	1	64Hz
1	0	32768Hz
1	1	8051 的工作频率

TSS21 TSS20 Timer2 Clock source

0	0	4Hz
0	1	64Hz
1	0	32768Hz
1	1	8051 的工作频率

ITS11 ITS10 Timer1 Clock source

0	0	8051Pin T1 接 4Hz
0	1	8051Pin T1 接 64Hz
1	0	8051Pin T1 接 32768Hz
1	1	8051 的工作频率/8

ITS01 ITS00 Timer0 Clock source

0	0	8051Pin T0 接 4Hz
0	1	8051Pin T0 接 64Hz
1	0	8051Pin T0 接 32768Hz
1	1	8051 的工作频率/8

•8849H --- Watch Dog Timer Reset/时钟脉冲控制寄存器 Read/Write

	ENWD	WDT1	WDT0	--	--	T2	T1	T0
Reset	0	0	0	--	--	0	0	0

读此寄存器会清零 Watch Dog Timer。

ENWD: 在正常工作模式, 此位对 Watchdog 没有影响, Watchdog 只工作在 enable 状态;
在软件开发模式, 此位可以读写, 并且影响 Watchdog:

= 0: disable watchdog = 1: enable watchdog

WDT[1-0]: Watch Dog timer select

WDT[1-0]	Watch Dog Length
0 0	62.5 ms
0 1	125 ms
1 0	250 ms
1 1	2 s

T2	T1	T0	(决定 TIMEOUT 管脚的输出频率)
0	0	0	1Hz 脉冲输出——(1S)
0	0	1	2Hz 脉冲输出——(0.5S)
0	1	0	4Hz 脉冲输出——(0.25S)
0	1	1	8Hz 脉冲输出——(0.125S)
1	0	0	32Hz 脉冲输出——(31.25ms)
1	0	1	128Hz 脉冲输出——(7.8125ms)
1	1	0	1024Hz 脉冲输出
1	1	1	32768Hz 脉冲输出 (用于时钟校验)

注意: 其中 1/2/4/8/32Hz 受实时调校寄存器 8806H 的影响, 1024/32768Hz 不受实时调校寄存器的影响。

•884AH --- 显示方式寄存器 Read/Write

	LCD1	LCD0	SPC1	SPC0	D/NC	DisMode2	DisMode1	DisMode0
Reset	0	0	0	0	0	0	0	0

D/NC: 当用两相两拍计数器输出时, 作为 F1、F2 和 F3、F4 切换的控制信号。当用两相四拍计数器输出时, 作为 FA1、FA2、FA3、FA4 和 FB1、FB2、FB3、FB4 切换的控制信号。

0 两相两拍时, 脉冲只输出到 F1、F2; 两相四拍时, 脉冲只输出到 FA1、FA2、FA3、FA4。

1 两相两拍时, 脉冲只输出到 F3、F4; 两相四拍时, 脉冲只输出到 FB1、FB2、FB3、FB4。

LCD1 LCD0 DisMode2 DisMode1 DisMode0

管脚复用方式

0	0	0	0	0	4×24LCD, Pin78~85、86~94 用作 GPIO PortD、E
0	1	0	0	0	4×32LCD, Pin86~94 用作 GPIO PortE
1	0	0	0	0	4×32LCD, Pin78~85 用作 GPIO PortD
1	1	0	0	0	4×40LCD
0	0	0	0	1	LED 方式, Pin52~67 用作位、段扫描输出, 其他作为 GPIO PortC/D/E/F
0	0	0	1	0	计数器两相两拍输出方式, Pin95~98 用作 F1~F4, Pin52~94 用作 GPIO PortA/B/C/D/E
0	0	0	1	1	计数器两相四拍输出方式, Pin86~89、91~94 用作 FA1、FA2、FA3、FA4、FB1、FB2、FB3、FB4, 其他作 PortA/B/C/D/F
0	0	1	X	X	I/O 模式, Pin52~98 全部用作 GPIO PortA/B/C/D/E/F

注意：只有在（00010）和（00011）两种情况下，D/NC才有效。

8051的工作频率由6M的时钟频率分频而来。

SPC1	SPC0	
0	0	分频系数为1（6M）
0	1	分频系数为2（3M）
1	0	分频系数为4（1.5M）

•8850H --- Port A I/O Control Read/Write

	CPA7	CPA6	CPA5	CPA4	CPA3	CPA2	CPA1	CPA0
RESET	0	0	0	0	0	0	0	0

CPAn = 0, 输出; = 1, 输入。

•8851H --- Port A Data Read/Write

	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
RESET	0	0	0	0	0	0	0	0

设置为输入 Pin 时，读入的是 Pin 上的状态；此时写无效。

设置为输出 Pin 时，写入的数据会输出至 Pin 上。

•8852H --- Port B I/O Control Read/Write

	CPB7	CPB6	CPB5	CPB4	CPB3	CPB2	CPB1	CPB0
RESET	0	0	0	0	0	0	0	0

CPBn = 0, 输出; = 1, 输入。

•8853H --- Port B Data Read/Write

	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
RESET	0	0	0	0	0	0	0	0

设置为输入 Pin 时，读入的是 Pin 上的状态；此时写无效。

设置为输出 Pin 时，写入的数据会输出至 Pin 上。

•8854H --- Port C I/O Control Read/Write

	CPC7	CPC6	CPC5	CPC4	CPC3	CPC2	CPC1	CPC0
RESET	0	0	0	0	0	0	0	0

CPCn = 0, 输出; = 1, 输入。

•8855H --- Port C Data Read/Write

	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
RESET	0	0	0	0	0	0	0	0

设置为输入 Pin 时，读入的是 Pin 上的状态；此时写无效。

设置为输出 Pin 时，写入的数据会输出至 Pin 上。

•8856H --- Port D I/O Control Read/Write

	CPD7	CPD6	CPD5	CPD4	CPD3	CPD2	CPD1	CPD0
RESET	0	0	0	0	0	0	0	0

CPDn = 0, 输出; = 1, 输入。

•8857H --- Port D Data Read/Write

	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
RESET	0	0	0	0	0	0	0	0

设置为输入 Pin 时, 读入的是 Pin 上的状态; 此时写无效。

设置为输出 Pin 时, 写入的数据会输出至 Pin 上。

•8858H --- Port E I/O Control Read/Write

	CPE7	CPE6	CPE5	CPE4	CPE3	CPE2	CPE1	CPE0
RESET	0	0	0	0	0	0	0	0

CPEn = 0, 输出; = 1, 输入。

•8859H --- Port E Data Read/Write

	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
RESET	0	0	0	0	0	0	0	0

设置为输入 Pin 时, 读入的是 Pin 上的状态; 此时写无效。

设置为输出 Pin 时, 写入的数据会输出至 Pin 上。

•885AH --- Port F I/O Control Read/Write

	---	---	---	---	CPF3	CPF2	CPF1	CPF0
RESET	---	---	---	---	0	0	0	0

CPF_n = 0, 输出; = 1, 输入。

•885BH --- Port F Data Read/Write

	---	---	---	---	PF3	PF2	PF1	PF0
RESET	---	---	---	---	0	0	0	0

设置为输入 Pin 时, 读入的是 Pin 上的状态; 此时写无效。

设置为输出 Pin 时, 写入的数据会输出至 Pin 上。

•885CH --- Port G I/O Control Read/Write

	CPG7	CPG6	CPG5	CPG4	CPG3	CPG2	CPG1	CPG0
RESET	0	0	0	0	0	0	0	0

CPG_n = 0, 输出; = 1, 输入。

•885DH --- Port G Data Read/Write

	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
RESET	0	0	0	0	0	0	0	0

设置为输入 Pin 时, 读入的是 Pin 上的状态; 此时写无效。

设置为输出 Pin 时, 写入的数据会输出至 Pin 上。

•885EH --- Port H I/O Control Read/Write

	---	---	---	---	CPH3	CPH2	CPH1	CPH0
RESET	---	---	---	---	0	0	0	0

注意：Watchdog 复位会对 ATT7023 内部所有有初试值的寄存器进行复位。通过 88D1H 进行软件复位只会将计量部分的数据寄存器复位。

QRun/Stop: = 0, 无功进入计量状态 = 1, 无功停止计量

PRun/Stop: = 0, 有功进入计量状态 = 1, 有功停止计量

注意：有功和无功能量停止计量时，功率和有效值等寄存器仍然有效

窃电标识/反向有功功率标识寄存器 88D4H

	--	--	--	QMODE	Pmode	--	Revp	Fault
Reset	--	--	--	0	0	--	0	0

Pmode=0: 有功电能累加采用绝对值方式，即不考虑正负性

=1: 有功电能累加采用向量方式，即考虑正负性，正加一，负减一

Qmode=0: 无功电能累加采用绝对值方式，即不考虑正负性

=1: 无功电能累加采用向量方式，即考虑正负性，正加一，负减一

Fault: Read/Write

Fault =0: 选择第一通道作为电流输入通道；

Fault =1: 选择第二通道作为电流输入通道。

Revp: 反向有功功率指示标识信号，当检测到负功率时，即电压和电流信号的相位差大于 90° 时，该信号为 1。当再次检测到正功率时，该信号为 0。 Read Only

HF 分频寄存器

High Byte 88D5H Read/Write

	---	---	---	---	---	HF10	HF09	HF08
Reset	---	---	---	---	---	0	0	1

Low Byte 88D6H Read/Write

	HF07	HF06	HF05	HF04	HF03	HF02	HF01	HF00
Reset	0	0	0	0	0	0	0	0

在稳定负载条件下，HF 输出频率与电能或有功功率成正比，这个输出频率可以向外部校准设备提供一个简单的光电隔离接口。此输出频率用 HF 分频寄存器来标定，该寄存器是一个可以对 $1\sim 2^{12}$ 输出频率范围进行标定的 11 位寄存器。

有功电能增益（通道一） Gp1

High Byte 88D7H Read/Write

	GPD15	GPD14	GPD13	GPD12	GPD11	GPD10	GPD09	GPD08
Reset	0	0	0	0	0	0	0	0

Low Byte 88D8H Read/Write

	GPD07	GPD06	GPD05	GPD04	GPD03	GPD02	GPD01	GPD00
Reset	0	0	0	0	0	0	0	0

无功电能增益（通道一） Gq1

High Byte 88D9H Read/Write

	GQD15	GQD14	GQD13	GQD12	GQD11	GQD10	GQD09	GQD08
--	-------	-------	-------	-------	-------	-------	-------	-------

Reset	0	0	0	0	0	0	0	0
-------	---	---	---	---	---	---	---	---

Low Byte **88DAH** Read/Write

	GQD07	GQD06	GQD05	GQD04	GQD03	GQD02	GQD01	GQD00
Reset	0	0	0	0	0	0	0	0

相位校正（通道一） **Gphs1**

High Byte **88DBH** Read/Write

	GPFD15	GPFD14	GPFD13	GPFD12	GPFD11	GPFD10	GPFD09	GPFD08
Reset	0	0	0	0	0	0	0	0

Low Byte **88DCH** Read/Write

	GPFD07	GPFD06	GPFD05	GPFD04	GPFD03	GPFD02	GPFD01	GPFD00
Reset	0	0	0	0	0	0	0	0

有功电能增益（通道二） **Gp2**

High Byte **88DDH** Read/Write

	GPD15	GPD14	GPD13	GPD12	GPD11	GPD10	GPD09	GPD08
Reset	0	0	0	0	0	0	0	0

Low Byte **88DEH** Read/Write

	GPD07	GPD06	GPD05	GPD04	GPD03	GPD02	GPD01	GPD00
Reset	0	0	0	0	0	0	0	0

无功电能增益（通道二） **Gq2**

High Byte **88DFH** Read/Write

	GQD15	GQD14	GQD13	GQD12	GQD11	GQD10	GQD09	GQD08
Reset	0	0	0	0	0	0	0	0

Low Byte **88E0H** Read/Write

	GQD07	GQD06	GQD05	GQD04	GQD03	GQD02	GQD01	GQD00
Reset	0	0	0	0	0	0	0	0

相位校正（通道二） **Gphs2**

High Byte **88E1H** Read/Write

	GPFD15	GPFD14	GPFD13	GPFD12	GPFD11	GPFD10	GPFD09	GPFD08
Reset	0	0	0	0	0	0	0	0

Low Byte **88E2H** Read/Write

	GPFD07	GPFD06	GPFD05	GPFD04	GPFD03	GPFD02	GPFD01	GPFD00
Reset	0	0	0	0	0	0	0	0

LF 分频寄存器 **88E3H** Read/Write

	LF7	LF6	LF5	LF4	LF3	LF2	LF1	LF0
Reset	0	0	1	0	0	0	0	0

通道一电流有效值 **I1rms**

High Byte **88E4H** Read Only

	I1D23	I1D22	I1D21	I1D20	I1D19	I1D18	I1D17	I1D16
Reset	0	0	0	0	0	0	0	0

Mid Byte **88E5H** Read Only

	I1D15	I1D14	I1D13	I1D12	I1D11	I1D10	I1D09	I1D08
Reset	0	0	0	0	0	0	0	0

Low Byte **88E6H** Read Only

	I1D07	I1D06	I1D05	I1D04	I1D03	I1D02	I1D01	I1D00
Reset	0	0	0	0	0	0	0	0

通道二电流有效值 I_{2rms} High Byte **88E7H** Read Only

	I2D23	I2D22	I2D21	I2D20	I2D19	I2D18	I2D17	I2D16
Reset	0	0	0	0	0	0	0	0

Mid Byte **88E8H** Read Only

	I2D15	I2D14	I2D13	I2D12	I2D11	I2D10	I2D09	I2D08
Reset	0	0	0	0	0	0	0	0

Low Byte **88E9H** Read Only

	I2D07	I2D06	I2D05	I2D04	I2D03	I2D02	I2D01	I2D00
Reset	0	0	0	0	0	0	0	0

电压有效值 U_{rms} High Byte **88EAH** Read Only

	UD23	UD22	UD21	UD20	UD19	UD18	UD17	UD16
Reset	0	0	0	0	0	0	0	0

Mid Byte **88EBH** Read Only

	UD15	UD14	UD13	UD12	UD11	UD10	UD09	UD08
Reset	0	0	0	0	0	0	0	0

Low Byte **88ECH** Read Only

	UD07	UD06	UD05	UD04	UD03	UD02	UD01	UD00
Reset	0	0	0	0	0	0	0	0

有功功率波形 P High Byte **88EDH** Read Only

	PD23	PD22	PD21	PD20	PD19	PD18	PD17	PD16
Reset	0	0	0	0	0	0	0	0

Mid Byte **88EEH** Read Only

	PD15	PD14	PD13	PD12	PD11	PD10	PD09	PD08
Reset	0	0	0	0	0	0	0	0

Low Byte **88EFH** Read Only

	PD07	PD06	PD05	PD04	PD03	PD02	PD01	PD00
Reset	0	0	0	0	0	0	0	0

无功功率波形 Q

High Byte **88F0H** Read Only

	QD23	QD22	QD21	QD20	QD19	QD18	QD17	QD16
Reset	0	0	0	0	0	0	0	0

Mid Byte **88F1H** Read Only

	QD15	QD14	QD13	QD12	QD11	QD10	QD09	QD08
Reset	0	0	0	0	0	0	0	0

Low Byte **88F2H** Read Only

	QD07	QD06	QD05	QD04	QD03	QD02	QD01	QD00
Reset	0	0	0	0	0	0	0	0

有功能量 Ep

Byte4 **88F3H** Read Only

	EPD39	EPD38	EPD37	EPD36	EPD35	EPD34	EPD33	EPD32
Reset	0	0	0	0	0	0	0	0

Byte3 **88F4H** Read Only

	EPD31	EPD30	EPD29	EPD28	EPD27	EPD26	EPD25	EPD24
Reset	0	0	0	0	0	0	0	0

Byte2 **88F5H** Read Only

	EPD23	EPD22	EPD21	EPD20	EPD19	EPD18	EPD17	EPD16
Reset	0	0	0	0	0	0	0	0

Byte1 **88F6H** Read Only

	EPD15	EPD14	EPD13	EPD12	EPD11	EPD10	EPD09	EPD08
Reset	0	0	0	0	0	0	0	0

Byte0 **88F7H** Read Only

	EPD07	EPD06	EPD05	EPD04	EPD03	EPD02	EPD01	EPD00
Reset	0	0	0	0	0	0	0	0

无功能量 Eq

Byte4 **88F8H** Read Only

	EQD39	EQD38	EQD37	EQD36	EQD35	EQD34	EQD33	EQD32
Reset	0	0	0	0	0	0	0	0

Byte3 **88F9H** Read Only

	EQD31	EQD30	EQD29	EQD28	EQD27	EQD26	EQD25	EQD24
Reset	0	0	0	0	0	0	0	0

Byte2 **88FAH** Read Only

	EQD23	EQD22	EQD21	EQD20	EQD19	EQD18	EQD17	EQD16
Reset	0	0	0	0	0	0	0	0

Byte1 **88FBH** Read Only

	EQD15	EQD14	EQD13	EQD12	EQD11	EQD10	EQD09	EQD08
Reset	0	0	0	0	0	0	0	0

Byte0 **88FCH** Read Only

	EQD07	EQD06	EQD05	EQD04	EQD03	EQD02	EQD01	EQD00
Reset	0	0	0	0	0	0	0	0

频率 Freq

High Byte **88FDH** Read Only

	FD7	FD6	FD5	FD4	FD3	FD2	FD1	FD0
Reset	1	1	1	1	1	1	1	1

Low Byte **88FEH** Read Only

	FD7	FD6	FD5	FD4	FD3	FD2	FD1	FD0
Reset	1	1	1	1	1	1	1	1

附录 2: 8051 相关资料

一、8051 的指令系统

指令类型		指令格式	字节数	指令周期	机器码
数据传送类指令	通用传送指令	MOV A, #data	2	2	74
		MOV data, #data	3	3	75
		MOV @Ri, #data	2	2	76-77
		MOV Rn, #data	2	2	78-7F
		MOV A, data	2	2	E5
		MOV A, @Ri	1	1	E6-E7
		MOV A, Rn	1	1	E8-EF
		MOV data, A	2	2	F5
		MOV @Ri, A	1	1	F6-F7
		MOV Rn, A	1	1	F8-FF
		MOV data, data	3	3	85
		MOV data, @Ri	2	2	86-87
		MOV data, Rn	2	2	8-8F
		MOV @Ri, data	2	2	A6-A7
		MOV Rn, data	2	2	A8-AF
		MOV DPTR, #data16	3	3	90
	外部 RAM(或 I/O 口)与累加器 A 传送指令	MOVX A, @DPTR	1	2-9	E0
		MOVX A, @Ri	1	2-9	E2-E3
		MOVX @DPTR, A	1	2-9	F0
		MOVX @Ri, A	1	2-9	F2-F3
	程序存储器向累加器 A 传送指令	MOVC A, @A+PC	1	3	83
		MOVC A, @A+DPTR	1	3	93
	数据交换指令	XCH A, data	2	2	C5
		XCH A, @Ri	1	1	C6-C7
		XCH A, Rn	1	1	C8-CF
		XCHD A, @Ri	1	1	D6-D7
栈操作指令	PUSH data	2	2	C0	
	POP data	2	2	D0	
逻辑操作类指令	对累加器 A 进行的逻辑操作	CLR A	1	1	E4
		CPL A	1	1	F4
		RL A	1	1	23
		RLC A	1	1	33
		RR A	1	1	03
		RRC A	1	1	13
	SWAP A	1	1	C4	
	双操作数指令	ANL(ORL, XRL) A, #data	2	2	54(44,64)
		ANL(ORL, XRL) A, data	2	2	55(45,65)
		ANL(ORL, XRL) A, @Ri	1	1	56-57(46-47,66-67)
		ANL(ORL, XRL) A, Rn	1	1	58-5F(48-4F,68-6F)
		ANL(ORL, XRL) data, A	2	2	52(42,62)

		ANL(ORL, XRL) data, #data	3	3	53(43,63)
算术运算类指令	加减运算指令	ADD A, #data	2	2	24
		ADD A, data	2	2	25
		ADD A, @Ri	1	1	26-27
		ADD A, Rn	1	1	28-2F
		ADDC A, #data	2	2	34
		ADDC A, data	2	2	35
		ADDC A, @Ri	1	1	36-37
		ADDC A, Rn	1	1	38-3F
		SUBB A, #data	2	2	94
		SUBB A, data	2	2	95
		SUBB A, @Ri	1	1	96-97
		SUBB A, Rn	1	1	98-9F
	乘除运算指令	MUL AB	1	5	A4
		DIV AB	1	5	84
	增量、减量指令	INC(DEC) A	1	1	04(14)
		INC(DEC) data	2	2	05(15)
		INC(DEC) @Ri	1	1	06-07(16-17)
		INC(DEC) Rn	1	1	08-0F(18-1F)
		INC DPTR	1	3	A3
	二—十进制调整指令	DA A	1	1	D4
位操作指令	位数据传送指令	MOV C, bit	2	2	A2
		MOV bit, C	2	2	92
	位状态控制指令	CLR bit	2	2	C2
		CLR C	1	1	C3
		CPL bit	2	2	B2
		CPL C	1	1	B3
		SETB bit	2	2	D2
		SETB C	1	1	D3
	位逻辑操作指令	ANL C, bit	2	2	82
		ANL C, /bit	2	2	B0
		ORL C, bit	2	2	72
		ORL C, /bit	2	2	A0
	位条件转移指令	JC rel	2	3	40
		JNC rel	2	3	50
		JB bit, rel	3	4	20
		JNB bit, rel	3	4	30
		JBC bit, rel	3	4	10
程序转移指令	LJMP addr16	3	4	02	
	AJMP addr11	2	3	01-E1	
	SJMP rel	2	3	80	
	JMP @A+DPTR	1	3	73	
	JZ rel	2	3	60	
	JNZ rel	2	3	70	
	CJNE A, #data, rel	3	4	B4	

控制转移类指令		CJNE A, data, rel	3	4	B5
		CJNE @Ri, #data, rel	3	4	B6-B7
		CJNE Rn, #data, rel	3	4	B8-BF
		DJNZ data, rel	3	4	D5
		DJNZ Rn, rel	2	3	D8-DF
	子程序调用和返回指令	LCALL addr16	3	4	12
		ACALL addr11	2	3	11-F1
		RET	1	4	22
		RETI	1	4	32
		NOP	1	1	00

说明:

标号	意义	标号	意义
A	累加器	Bit	直接寻址位
Rn	寄存器 R0-R7	#data	8 位立即数
Data	直接寻址的地址	#data16	16 位立即数
@Ri	R0 和 R1 的间接寻址	Addr16	16 位的目标地址
Rel	跳转指令中单字节的偏移量	Addr11	11 位的目标地址

二、特殊功能寄存器

寄存器名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	地址
SP	-	-	-	-	-	-	-	-	81H
DPL0	-	-	-	-	-	-	-	-	82H
DPH0	-	-	-	-	-	-	-	-	83H
DPL1	-	-	-	-	-	-	-	-	84H
DPH1	-	-	-	-	-	-	-	-	85H
DPS	×	×	×	×	×	×	×	SEL	86H
PCON	SMOD	×	×	×	×	×	×	×	87H
TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	88H
TMOD	GATE	C/T	M1	M0	GATE	C/T	M1	M0	89H
TLO	-	-	-	-	-	-	-	-	8AH
TL1	-	-	-	-	-	-	-	-	8BH
TH0	-	-	-	-	-	-	-	-	8CH
TH1	-	-	-	-	-	-	-	-	8DH
CKCON	×	×	×	T1M	T0M	MD2	MD1	MD0	8EH
MPAGE	-	-	-	-	-	-	-	-	92H
SCON	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	98H
SBUF	-	-	-	-	-	-	-	-	99H
IE	EA	×	×	ES	ET1	EX1	ET0	EX0	A8H
IP	×	×	×	PS	PT1	PX1	PT0	PX0	B8H
PSW	CY	AC	F0	RS1	RS0	OV	×	P	D0H
ACC	-	-	-	-	-	-	-	-	E0H
B	-	-	-	-	-	-	-	-	FOH

三、特殊功能寄存器的默认值

寄存器名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	地址
-------	------	------	------	------	------	------	------	------	----

SP	0	0	0	0	0	1	1	1	81H
DPL0	0	0	0	0	0	0	0	0	82H
DPH0	0	0	0	0	0	0	0	0	83H
DPL1	0	0	0	0	0	0	0	0	84H
DPH1	0	0	0	0	0	0	0	0	85H
DPS	0	0	0	0	0	0	0	0	86H
PCON	0	0	1	1	0	0	0	0	87H
TCON	0	0	0	0	0	0	0	0	88H
TMOD	0	0	0	0	0	0	0	0	89H
TLO	0	0	0	0	0	0	0	0	8AH
TL1	0	0	0	0	0	0	0	0	8BH
TH0	0	0	0	0	0	0	0	0	8CH
TH1	0	0	0	0	0	0	0	0	8DH
CKCON	0	0	0	0	0	0	0	1	8EH
MPAGE	0	0	0	0	0	0	0	0	92H
SCON	0	0	0	0	0	0	0	0	98H
SBUF	0	0	0	0	0	0	0	0	99H
IE	0	0	0	0	0	0	0	0	A8H
IP	1	0	0	0	0	0	0	0	B8H
PSW	0	0	0	0	0	0	0	0	D0H
ACC	0	0	0	0	0	0	0	0	E0H
B	0	0	0	0	0	0	0	0	F0H

附录 3: 本应用手册更新内容**2004/6/29 更新内容:**

1. 增加对通用 I/O 口的描述 (第十一章);
2. 对 UART1 口的使用作详细的补偿说明 (第七章);
3. 增加工作模式控制的介绍 (9.3);
4. Timer3 的寄存器 8841H 说明更新 (10.4)。

2004/7/26 更新内容:

1. 增加第 4 章: 系统时钟;
2. 增加 Idle Mode 编程指南 (9.2.4);
3. 更新应用线路图 (第十四章)。

2004/11/24 更新内容:

1. 更新应用线路图 (第十四章);
2. 增加 PCB 布线时注意事项 (第十四章)。
3. 增加电源供电系统的描述 (9.4)。

2004/12/31 更新内容:

- 增加 OTP ROM 出厂设置 (13.4)。