

IP5902 数据手册

目录

1. 产品简述	4
1.1 产品特性	4
1.2 简介及系统框图	5
1.3 引脚说明	7
2. 电气参数	9
2.1 极限参数	9
2.2 推荐工作条件	9
2.3 电气特性	9
3. MCU 与存储器	11
3.1 概述	11
3.1.1 MCU 指令集	11
3.2 程序存储空间	14
3.3 数据存储空间	15
3.4 特殊功能寄存器	15
3.5 烧录口 ICP	16
3.6 Register	16
4. 系统控制模块	17
4.1 正常工作状态与低功耗	17
4.2 唤醒源	17
4.2.1 唤醒源查询寄存器	17
4.3 唤醒配置的相关寄存器	17
4.4 待机说明	18
5. 中断	20
5.1 中断向量表	20
5.2 中断置起的相关寄存器	20
5.3 中断源配置相关寄存器	21
5.4 中断优先级	23
6. 系统时钟	25
6.1 高频时钟	25
6.2 低频时钟	25
6.3 时钟配置相关寄存器	25
7.I/O 端口	26
7.1 I/O 口功能	26
7.2 I/O 端口配置寄存器	26
7.2.1 I/O 端口通用配置寄存器	26
7.2.2 I/O 端口数据寄存器	27
7.2.3 I/O 端口复用配置寄存器	27
8.硬件按键 Key	29
8.1 Key 模块介绍	29
8.2 key 状态相关寄存器	29

9.定时器	30
9.1 看门狗定时器	30
9.2 Timer0/Timer1 定时器/计数器	31
9.2.1 Timer0 定时器/计数器相关寄存器	31
9.2.2 Timer1 定时器/计数器相关寄存器	32
9.3 PWM	34
9.3.1 PWM0 相关寄存器	34
9.3.2 PWM1 相关寄存器	34
10.模数转换（ADC）	36
10.1 ADC 模块介绍	36
10.2 ADC 相关寄存器汇总	36
10.2.1 ADC 配置寄存器	36
10.2.2 ADC_VIN 数据寄存器	38
10.2.3 ADC_ICHG 数据寄存器	38
10.2.4 ADC_VBAT 数据寄存器	39
10.2.5 ADC_VGP1 数据寄存器	39
10.2.6 ADC_VGP2 数据寄存器	40
10.2.7 ADC_VGP3 数据寄存器	40
10.2.8 ADC_VOUT 数据寄存器	40
10.2.9 ADC_VND 数据寄存器	41
11.Charger 模块	42
11.1 模块介绍	42
11.2 Charger 相关寄存器	42
12.Boost 模块	44
12.1 Boost 模块介绍	44
12.2 Boost 相关寄存器	44
12.2.1 Boost 状态相关寄存器	44
13.VND 路径 MOS 模块	45
13.1 VND 路径 MOS 介绍	45
13.2 VND 路径 MOS 相关寄存器	45
12.2.1 VND 路径 MOS 控制寄存器	45
14.典型应用原理图	46
14.封装信息	47
15.IC 丝印说明	48
16.责任及版权声明	49

1. 产品简述

1.1 产品特性

- **MCU**
 - ✧ 兼容 8051 指令集内核
 - ✧ 8K Bytes OTP 型程序存储器
 - ✧ 256 Bytes RAM 通用数据存储器，支持直接寻址、间接寻址等多种寻址方式
 - ✧ 系统时钟：低速 32KHz、高速 12MHz
 - ✧ 2*16bits 定时器：Timer0、Timer1
 - ✧ 2 个 PWM 复用定时器 Timer1、Timer2
 - ✧ 8 通道 10bit 位宽 ADC：4 路内置电压采样通道，3 路外部 GPIO 采样通道，1 路充电电流采样
 - ✧ IO 口资源：支持最多 8 个普通 IO 口
- **异步升压**
 - ✧ 最大输出功率 9W，如 9V/1A，5V/1.8A 等
 - ✧ 升压输出电压可调 4V~9V
- **充电**
 - ✧ 最大 800mA 线性充电，电流可调节 25mA/Step
 - ✧ 支持 4.20V、4.30V、4.35V 和 4.4V 电池
 - ✧ 支持自动再充电功能
 - ✧ 可反馈充电状态
- **低功耗**
 - ✧ 待机功耗 35 μ A
 - ✧ DeepSleep 待机功耗小于 10uA
- **BOM 极简**
 - ✧ 集成异步升压功率 MOS
 - ✧ 集成 1 个路径 MOS
- **多重保护、高可靠性**
 - ✧ 输出过流、短路保护
 - ✧ 输入过压、过充保护
 - ✧ 整机过温保护
 - ✧ ESD 4KV，VIN/VOUT 瞬态耐压高达 15V
- **封装方式 SOP16**

1.2 简介及系统框图

IP5902 是一款集成 9V 异步升压转换、锂电池充电管理及负端 NMOS 管的 8-bit MCU 芯片，为小型电子设备提供高集成度解决方案。

IP5902 内置兼容 8051 指令集 8-bit MCU，最高为 12MHz 主频时钟，内置 256 Bytes RAM 及 8K Bytes OTP ROM 为便携电子设备提供充足的编程空间。

IP5902 内置一个最高 9V 输出的异步升压 DC-DC，功率管内置，提供最大 1A@9V~2.2A@4V 输出电流。

IP5902 的线性充电提供最大 800mA 充电电流，可灵活配置充电电流。内置输入欠压、过压、过流、IC 过温和输入欠压智能调节充电电流功能，提供安全可靠的充电管理。

IP5902 内置负端 NMOS。

IP5902 内置 20uA/100uA 恒流源及 10-bit ADC，可实现自定义 NTC 功能。

IP5902 的高集成度与丰富功能，使其在应用时仅需极少的外围器件，并有效减小整体方案的尺寸，降低 BOM 成本，图 1 为系统框架图。

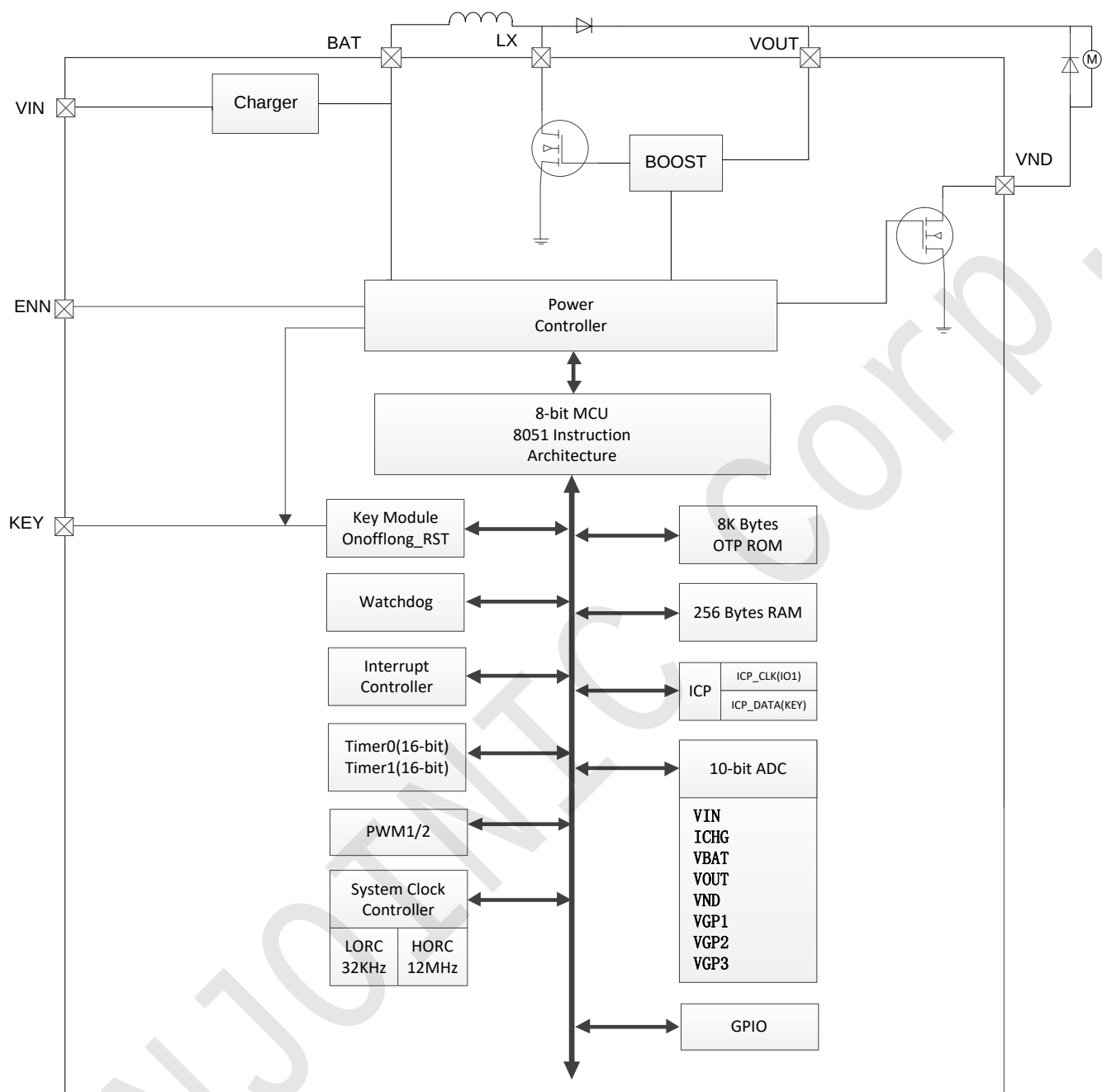


图 1 IP5902 内部系统框图

1.3 引脚说明



图 2 IP5902 引脚图

引脚编号	引脚名字	引脚功能			
		F1	F2	F3	F4
1	BAT	电池正端			
2	VIN	充电输入端			
3	ENN	EN 使能脚，低电平唤醒			
4	IO7	通用 IO7			
5	IO6	通用 IO6			
6	IO5	通用 IO5	PWM1H		
7	IO4	通用 IO4	PWM0	SIRQ	
8	IO3	通用 IO3	PWM0	ADC_VGP3	SIRQ
9	IO2	通用 IO2	PWM1L	ADC_VGP2	SIRQ
10	IO1	通用 IO1	PWM1H	ADC_VGP1	SIRQ
11	KEY	KEY 按键输入	通用 IO0		
12	VND	内部路径管正端			
13	LX	异步 BOOST 开关管脚			
14	GND	GND			
15	TEST	测试引脚，保持悬空			
16	VOUT	BOOST 输出电压检测脚			

1.4 修改记录

备注：以前版本的页码可能与当前版本的页码不同。

释放版本 V1.00（2023 年 8 月）	页码
• 初版释放.....	1
版本 V1.02 至 V1.03（2023 年 11 月）	页码
• 修改唤醒源配置相关寄存器的地址和名称.....	17
• 修改 ADC 相关寄存器的地址和名称.....	36
• 修改部分寄存器默认值.....	19
版本 V1.03 至 V1.04（2023 年 11 月）	页码
• 修改开关频率单位.....	10

2. 电气参数

2.1 极限参数

参数	符号	值	单位
高耐压引脚电压范围	VIN、VOUT、LX、VND	-0.3 ~ 15	V
低压引脚电压范围	IO0~IO7、BAT、ENN	-0.3 ~ 8	V
结温范围	T _J	-40 ~ 150	°C
存储温度范围	T _{stg}	-60 ~ 150	°C
热阻（结温到环境）	θ _{JA}	120	°C/W
人体模型（HBM）	ESD	4	KV

*高于绝对最大额定值部分所列数值的应力有可能对器件造成永久性的损害，在任何绝对最大额定值条件下暴露的时间过长都有可能影响器件的可靠性和使用寿命。

2.2 推荐工作条件

参数	符号	最小值	典型值	最大值	单位
输入电压	VIN	4.5	5	6.0	V
工作环境温度	T _A	-20	--	85	°C

*超出这些工作条件，器件工作特性不能保证。

2.3 电气特性

除特别说明，T_A=25°C，L=2.2uH

参数	符号	测试条件	最小值	典型值	最大值	单位
充电系统						
输入电压	VIN	V _{BAT} =3.7V	4.5	5	6.0	V
输入过压	VIN _{OV}		5.8	6	6.2	V
VIN 唤醒电压	VIN _{OK}		4.0	4.2	4.4	V
充电关闭电压	VIN _{UV}		4.0	4.2	4.4	V
CV 恒压充电电压	CV _{4.2V}		4.16	4.20	4.24	V
	CV _{4.30V}		4.28	4.30	4.34	V
	CV _{4.35V}		4.33	4.35	4.4	V
	CV _{4.4V}		4.38	4.40	4.44	V
充电截止电流	IVIN _{stop}	输入 VIN=5V	30	30	40	mA

充电电流	I_{VIN}	VIN 口充电电流输入端电流, VBAT=3.7V 设定充电电流为 350mA	320	350	380	mA
涓流充电电流	I_{TRKL}	VIN=5v, BAT=2.7v, 设定充电电流为 350mA	20	25	30	mA
涓流截止电压	V_{TRKL}		2.9	3	3.1	V
再充电阈值	V_{RCH}		4.00	4.05	4.10	V
升压系统						
电池工作电压	V_{BAT}		3.0		4.4	V
输出电压	V_{OUT}	VBAT=3.7V, Iout=100mA, 设置输出 5V	4.75	5.0	5.25	V
		VBAT=3.7V, Iout=100mA, 设置输出 7V	6.65	7.0	7.35	V
		VBAT=3.7V, Iout=100mA, 设置输出 9V	8.55	9.0	9.45	V
升压系统供电电流	I_{OUT}				1	A
负载过流检测时间	T_{UVD}	输出电压持续低于 3.6V		2		ms
负载短路检测时间	T_{OCD}	输出电流持续大于 2.0A		10		us
控制系统						
开关频率	f_s	放电开关频率		700		kHz
VND NMOS 导通电阻	R_{DSON}			135		mΩ
LX NMOS 导通电阻				100		mΩ
VCC 电压	VCC	VCC 无负载。VCC=VBAT。VBAT 浮空只有 VIN 供电时, VCC=3.3V	VBAT-0.1	VBAT	VBAT	V
电池待机电流	I_{STB_STOP}	VIN=0V, VBAT=3.7V, STOP 模式		35	45	uA
	I_{STB_DS}	VIN=0V, VBAT=3.7V, DeepSleep 模式		8	10	uA
IO 口驱动电流	I_{Gpio}		4	6	8	mA
短按按键时间	$T_{OnDebounce}$		100		300	ms
长按按键时间	$T_{Keylight}$		2		3	s
热关断结温	T_{OTP}	上升温度	130	140	150	°C
热关断迟滞	ΔT_{OTP}		30	40	50	°C

*上述数据基于 demo 程序测试, 多个参数规格可软件配置

3. MCU 与存储器

3.1 概述

3.1.1 MCU 指令集

- IP5902 内置 8-bit MCU，其指令集兼容 8051。

算数操作指令

Mnemonic	Description	Code	Bytes	Cycles
ADD A,Rn	Add register to accumulator	28-2F	1	1
ADD A,direct	Add direct byte to accumulator	25	2	2
ADD A,@Ri	Add indirect RAM to accumulator	26-27	1	2
ADD A,#data	Add immediate data to accumulator	24	2	2
ADDC A,Rn	Add register to accumulator with carry flag	38-3F	1	1
ADDC A,direct	Add direct byte to A with carry flag	35	2	2
ADDC A,@Ri	Add indirect RAM to A with carry flag	36-37	1	2
ADDC A,#data	Add immediate data to A with carry flag	34	2	2
SUBB A,Rn	Subtract register from A with borrow	98-9F	1	1
SUBB A,direct	Subtract direct byte from A with borrow	95	2	2
SUBB A,@Ri	Subtract indirect RAM from A with borrow	96-97	1	2
SUBB A,#data	Subtract immediate data from A with borrow	94	2	2
INC A	Increment accumulator	04	1	1
INC Rn	Increment register	08-0F	1	2
INC direct	Increment direct byte	05	2	3
INC @Ri	Increment indirect RAM	06-07	1	3
INC DPTR	Increment data pointer	A3	1	1
DEC A	Decrement accumulator	14	1	1
DEC Rn	Decrement register	18-1F	1	2
DEC direct	Decrement direct byte	15	2	3
DEC @Ri	Decrement indirect RAM	16-17	1	3
MUL AB	Multiply A and B	A4	1	5
DIV	Divide A by B	84	1	5
DA A	Decimal adjust accumulator	D4	1	1

逻辑操作指令

Mnemonic	Description	Code	Bytes	Cycles
ANL A,Rn	AND register to accumulator	58-5F	1	1
ANL A,direct	AND direct byte to accumulator	55	2	2
ANL A,@Ri	AND indirect RAM to accumulator	56-57	1	2
ANL A,#data	AND immediate data to accumulator	54	2	2
ANL direct,A	AND accumulator to direct byte	52	2	3
ANL direct,#data	AND immediate data to direct byte	53	3	4
ORL A,Rn	OR register to accumulator	48-4F	1	1
ORL A,direct	OR direct byte to accumulator	45	2	2
ORL A,@Ri	OR indirect RAM to accumulator	46-47	1	2
ORL A,#data	OR immediate data to accumulator	44	2	2
ORL direct,A	OR accumulator to direct byte	42	2	3
ORL direct,#data	OR immediate data to direct byte	43	3	4
XRL A,Rn	Exclusive OR register to accumulator	68-6F	1	1
XRL A,direct	Exclusive OR direct byte to accumulator	65	2	2
XRL A,@Ri	Exclusive OR indirect RAM to accumulator	66-67	1	2
XRL A,#data	Exclusive OR immediate data to accumulator	64	2	2
XRL direct,A	Exclusive OR accumulator to direct byte	62	2	3
XRL direct,#data	Exclusive OR immediate data to direct byte	63	3	4
CLR A	Clear accumulator	E4	1	1
CPL A	Complement accumulator	F4	1	1
RL A	Rotate accumulator left	23	1	1
RLC A	Rotate accumulator left through carry	33	1	1
RR A	Rotate accumulator right	03	1	1
RRC A	Rotate accumulator right through carry	13	1	1
SWAP A	Swap nibbles within the accumulator	C4	1	1

数据传输指令

Mnemonic	Description	Code	Bytes	Cycles
MOV A,Rn	Move register to accumulator	E8-EF	1	1
MOV A,direct	Move direct byte to accumulator	E5	2	2
MOV A,@Ri	Move indirect RAM to accumulator	E6-E7	1	2
MOV A,#data	Move immediate data to accumulator	74	2	2
MOV Rn,A	Move accumulator to register	F8-FF	1	2
MOV Rn,direct	Move direct byte to register	A8-AF	2	4
MOV Rn,#data	Move immediate data to register	78-7F	2	2
MOV direct,A	Move accumulator to direct byte	F5	2	3
MOV direct,Rn	Move register to direct byte	88-8F	2	3
MOV direct1,direct2	Move direct byte to direct byte	85	3	4
MOV direct,@Ri	Move indirect RAM to direct byte	86-87	2	4
MOV direct,#data	Move immediate data to direct byte	75	3	3
MOV @Ri,A	Move accumulator to indirect RAM	F6-F7	1	3
MOV @Ri,direct	Move direct byte to indirect RAM	A6-A7	2	5
MOV @Ri,#data	Move immediate data to indirect RAM	76-77	2	3
MOV DPTR,#data16	Load data pointer with a 16-bit constant	90	3	3
MOVC A,@A+DPTR	Move code byte relative to DPTR to accumulator	93	1	3
MOVC A,@A+PC	Move code byte relative to PC to accumulator	83	1	3
MOVX A,@Ri	Move external RAM (8-bit addr.) to A	E2-E3	1	3-10
MOVX A,@DPTR	Move external RAM (16-bit addr.) to A	E0	1	3-10
MOVX @Ri,A	Move A to external RAM (8-bit addr.)	F2-F3	1	4-11
MOVX @DPTR,A	Move A to external RAM (16-bit addr.)	F0	1	4-11
PUSH direct	Push direct byte onto stack	C0	2	4
POP direct	Pop direct byte from stack	D0	2	3
XCH A,Rn	Exchange register with accumulator	C8-CF	1	2
XCH A,direct	Exchange direct byte with accumulator	C5	2	3
XCH A,@Ri	Exchange indirect RAM with accumulator	C6-C7	1	3
XCHD A,@Ri	Exchange low-order nibble indir. RAM with A	D6-D7	1	3

位操作指令

Mnemonic	Description	Code	Bytes	Cycles
CLR C	Clear carry flag	C3	1	1
CLR bit	Clear direct bit	C2	2	3
SETB C	Set carry flag	D3	1	1
SETB bit	Set direct bit	D2	2	3
CPL C	Complement carry flag	B3	1	1
CPL bit	Complement direct bit	B2	2	3
ANL C,bit	AND direct bit to carry flag	82	2	2
ANL C,/bit	AND complement of direct bit to carry	B0	2	2
ORL C,bit	OR direct bit to carry flag	72	2	2
ORL C,/bit	OR complement of direct bit to carry	A0	2	2
MOV C,bit	Move direct bit to carry flag	A2	2	2
MOV bit,C	Move carry flag to direct bit	92	2	3

程序跳转指令

Mnemonic	Description	Code	Bytes	Cycles
ACALL addr11	Absolute subroutine call	xxx11	2	6
LCALL addr16	Long subroutine call	12	3	6
RET	from subroutine	22	1	4
RETI	from interrupt	32	1	4
AJMP addr11	Absolute jump	xxx01	2	3
LJMP addr16	Long jump	02	3	4
SJMP rel	Short jump (relative addr.)	80	2	3
JMP @A+DPTR	Jump indirect relative to the DPTR	73	1	2
JZ rel	Jump if accumulator is zero	60	2	3
JNZ rel	Jump if accumulator is not zero	70	2	3
JC rel	Jump if carry flag is set	40	2	3
JNC	Jump if carry flag is not set	50	2	3
JB bit,rel	Jump if direct bit is set	20	3	4
JNB bit,rel	Jump if direct bit is not set	30	3	4
JBC bit,direct rel	Jump if direct bit is set and clear bit	10	3	4
CJNE A,direct rel	Compare direct byte to A and jump if not equal	B5	3	4
CJNE A,#data rel	Compare immediate to A and jump if not equal	B4	3	4
CJNE Rn,#data rel	Compare immed. to reg. and jump if not equal	B8-BF	3	4
CJNE @Ri,#data rel	Compare immed. to ind. and jump if not equal	B6-B7	3	4
DJNZ Rn,rel	Decrement register and jump if not zero	D8-DF	2	3
DJNZ direct,rel	Decrement direct byte and jump if not zero	D5	3	4
NOP	No operation	00	1	1

3.2 程序存储空间

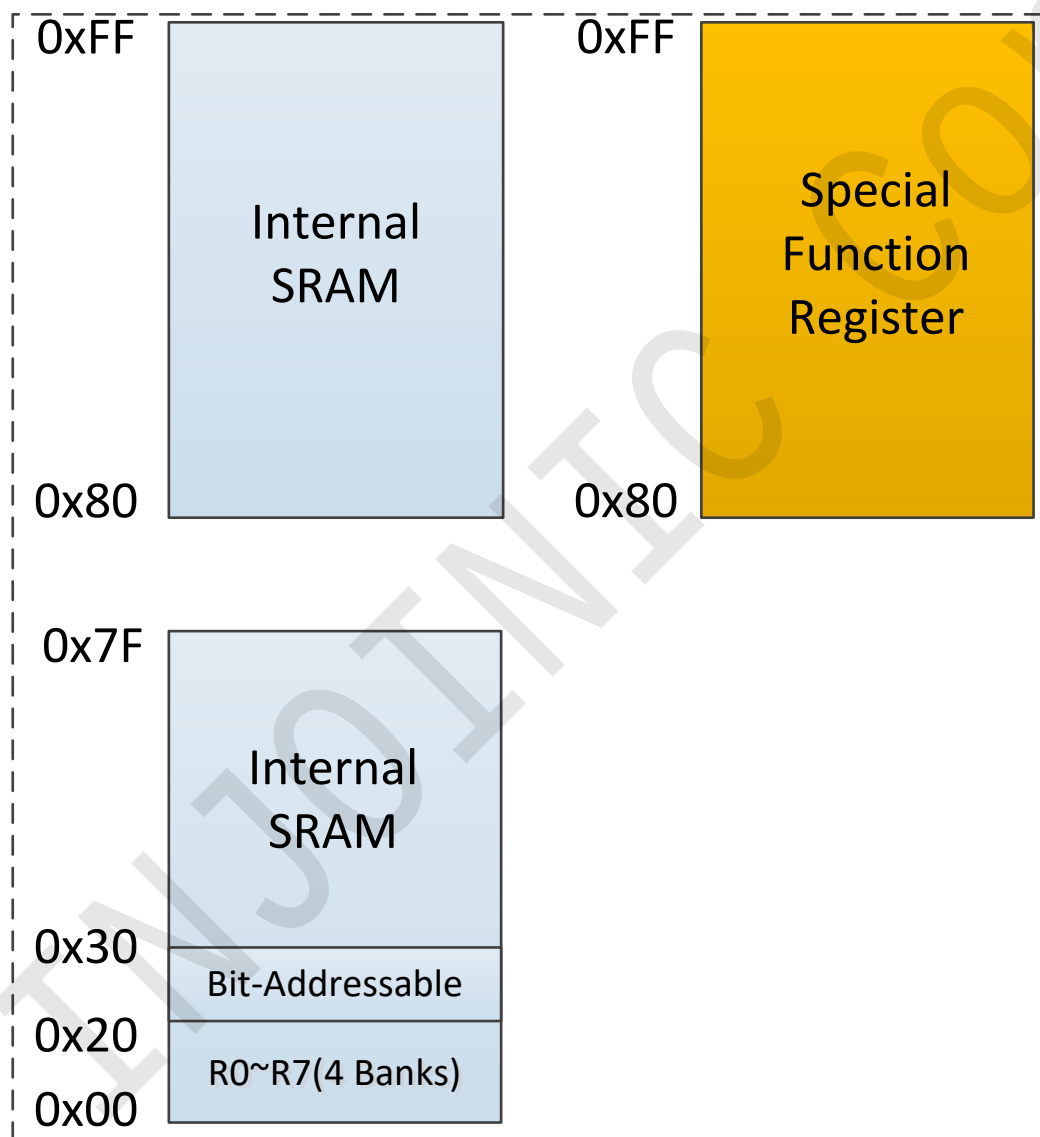
- IP5902 内置 8K Bytes OTP 程序存储空间，用来存放用户代码及储存程序。
- 程序存储空间由程序计数器来寻址，也包含数据、表格、中断入口等。
- 程序存储空间内的某些地址用作诸如复位和中断入口等特殊用途。

3.3 数据存储空间

- 内部 SRAM 空间共 256 Bytes，地址 00h~FFh。
- 其中低 128 Bytes（地址 0x00~0x7F）支持直接和间接寻址方式，高 128 Bytes（地址 0x80~0xFF）只支持间接寻址方式。

3.4 特殊功能寄存器

- SFR 特殊寄存器空间地址占据数据空间高 128 Bytes（地址 0x80~0xFF）只支持间接寻址方式。



3.5 烧录口 ICP

程序烧录口包含 2 根信号线：ICP_DATA(复用 KEY)和 ICP_CLK(复用 GPIO1)。

上电的时候，若检测 KEY/GPIO1 均为高电平并持续大于 5ms，则将 KEY/GPIO1 切换为 ICP 接口，否则 KEY/GPIO1 当做正常功能使用。

3.6 Register

CPU 的内部寄存器有如下几个。

序号	寄存器符号	寄存器地址	寄存器名称和描述
1	SP	81h	Stack Pointer
2	DPL	82h	Data Pointer Low 0
3	DVM	83h	Data Pointer High 0
4	DPL1	84h	Data Pointer Low 1
5	DVM1	85h	Data Pointer High 1
6	PCON	87h	MCU Power Control
7	CKCON	8Eh	MCU Clock Control
8	DPS	92h	Data Pointer Select Register
9	PSW	D0h	Program status word
10	ACC	E0h	Accumulator
11	B	F0h	B Register

4. 系统控制模块

4.1 正常工作状态与低功耗

- 芯片正常工作状态下，所有时钟均使能。
- 芯片的休眠模式即为低功耗模式，仅 LOSC 时钟工作。
- 执行 STOP 指令可使系统进入低功耗模式。

4.2 唤醒源

IP5902 系统的 Wakeup 信号源包括：

- 1、5V 适配器接入 VIN；
- 2、Key 按下（包括短按/长按/上升沿/下降沿等，任选一种）；
- 3、外部中断 SIRQ；
- 4、Watchdog；

唤醒 IP5902 后，用户可以通过只读寄存器 DEBUG6(0x8F)对应的 bit 位来查询唤醒方式。

IP5902 第一次 BAT 接电池无需充电激活：ENN 脚接地使能后，第一次 BAT 接电池后，按键就能唤醒输出；

4.2.1 唤醒源查询寄存器

DEBUG6(0x8F)

Offset = 0x8F, Default=0x00

Bit(s)	Name	Description	R/W	Reset
7	WK_FLAG[7]	唤醒源 vinwk	R	0
6	WK_FLAG[6]	唤醒源 sirq_wk	R	0
5	WK_FLAG[5]	唤醒源 wdg_rst	R	0
4	WK_FLAG[4]	唤醒源按键唤醒	R	0
3		—	R	0
2	WK_FLAG[2]	唤醒源按键长按复位 rst	R	0
1		—	R	0
0		—	R	0

4.3 唤醒配置的相关寄存器

WK_CTL1(0x94)

Offset = 0x94, Default=0x00

Bit(s)	Name	Description	R/W	Reset
7	ONOFFRST_EN	超长按强制复位使能 0: enable 1: disable	R/W	0x0
6:3	—	Reserved	R/W	0
2:0	SIRQ_WAKEUP_SET	Sirq 唤醒选择：	R/W	000

		000: disable 001: 上升沿 010: 下降沿 011: 上升/下降沿 100: 低电平 101: 高电平		
--	--	---	--	--

WK_CTL0(0x98)

Offset = 0x98, Default = 0xC0

Bit(s)	Name	Description	R/W	Reset
7	S2OFF_EN	双击关机使能 1: enable, 双击时屏蔽单击信号 0: disable, 双击时同时存在单击信号	R/W	1
6:4	ONOFF_WKUPEN_MODE	按键唤醒选择: 000: Disable 001: 抬起来 010: 按下去 011: 上升沿+下降沿 100: 短按 101: 长按 110-111: Reserved	R/W	100
3	BATLOW_OFF_EN	Batlow 关机信号是否有效 1: enable 0: disable		0
2	-	Reserved		
1	ONOFF_LONG_RST_WK_EN	配置为 Onoff 超长按产生复位时, 是否在该复位信号撤销时, 重启系统。 1: enable 0: disable	R/W	0
0	WDG_RST_WK_EM	配置 wdg 发出 rst 后的唤醒使能: 1: enable 0: disable	R/W	0

4.4 待机说明

IP5902 有 2 种待机状态:

- 1、stop 状态, 待机功耗 35uA, 可以通过按键、外部 SIRQ 等唤醒;
 - 2、deepsleep 状态, 待机功耗小于 10uA, 无法通过按键、外部 SIRQ 等唤醒, 只能通过插入 VIN 或 ENN 脚拉低来唤醒;
- 可以通过 STOP_EN(0x99)寄存器来进入 stop 状态 (0x99=0x01);
 通过 Deepsleep_EN(0xEC)寄存器来进入 deepsleep 状态 (0xEC=0x4E);

STOP_EN(0x99)

Addr = 0x99 Default = 0x00

bit	Name	Description	R/W	Reset
7:1	--	Reserved	RW	0
0	STOP_EN	写 1 进 stop。	W	0

Deepsleep_EN(0xEC)

Addr = 0xEC Default = 0x66

bit	Name	Description	R/W	Reset
7:6	--	Reserved	RW	01
5	DEEPSLEEP_EN	写 0 进 deepsleep。	RW	1
4:0	--	Reserved	RW	00110

5. 中断

IP5902 系统的中断源包括：

- 1、Boost 异常中断：欠压 LOWP 或短路 SCDT
- 2、电池低电 Batlow_irq（2.8/2.9/3.0/3.1 四挡可选）
- 3、Key（包括长按键、双击、短按键、抬起来、按下去）
- 4、WatchDog
- 5、Timer0 或 Timer1
- 6、IN 过压 VINOV，VIN 欠压 VINUV，VIN 插入 VIN_IN，VIN 拔出 VIN_OUT
- 7、外部 SIRQ
- 8、过温 OT

5.1 中断向量表

中断序号	中断名称	中断向量
0	SCDT Lowpot	0x03
1	Watchdog	0x0b
2	Timer0	0x13
3	Timer1	0x1b
4	SIRQ	0x23
5	Batlow_irq	0x2b
6	VIN_IN VIN_OUT	0x33
7	NC	0x3b
8	NC	0x43
9	NC	0x4b
10	KEY	0x53
11	NC	0x5b
12	NC	0x63
13	VINOV VINUV	0x6b
14	NC	0x73
15	OT	0x7b
16	NC	0x83

5.2 中断置起的相关寄存器

IRQ_PEND0(0xA1)

Offset = 0xA1, Default=0x00

Bit(s)	Name	Description	R/W	Reset
7	BST_SCDT_IRQ	Boost 异常 SCDT（写 1 清除）	R/W	0
6	BST_LOWP_IRQ	Boost 异常 LOWP（写 1 清除）	R/W	0
5	BATLOW_IRQ	Batlow（写 1 清除）	R/W	0
4	LongKey_IRQ	长按键（写 1 清除）	R/W	0

3	DoubleKey_IRQ	双击（写 1 清除）	R/W	0
2	ShortKey_IRQ	短按键（写 1 清除）	R/W	0
1	KeyUp_IRQ	按键抬起来（写 1 清除）	R/W	0
0	KeyDown_IRQ	按键按下去（写 1 清除）	R/W	0

IRQ_PEND1(0xA2)

Offset = 0xA2, Default=0x11

Bit(s)	Name	Description	R/W	Reset
7	WDG_IRQ_PEND	Watchdog 中断标志（写 1 清除）	R/W	0
6	TIMER1_IRQ_PEND	Timer1 中断标志（写 1 清除）	R/W	0
5	TIMER0_IRQ_PEND	Timer0 中断标志（写 1 清除）	R/W	0
4	—	reserved	R/W	1
3	VINUV_IRQ_PEND	VIN 欠压中断标志，寄存器配置的欠压阈值 4.7/4.6/4.25 可选（写 1 清除）	R/W	0
2	VINOV_IRQ_PEND	Vin 过压中断标志（写 1 清除）	R/W	0
1	VIN_OUT_IRQ_PEND	Vin 拔出中断标志，vinok 掉下去的阈值，3.6V/3.1V 可选（写 1 清除）	R/W	0
0	VIN_IN_IRQ_PEND	Vin 插入中断标志（写 1 清除）	R/W	1

IRQ_PEND2(0xA3)

Offset = 0xA3, Default=0x00

Bit(s)	Name	Description	R/W	Reset
7	OT_IRQ_PEND	OT 过温	R/W	0
6:3	—	Reserved	R/W	0
2	SIRQ_PEND	SIRQ 中断	R/W	0
1:0	—	Reserved	R/W	0

5.3 中断源配置相关寄存器

IRQ_CTL(0xA0)

Offset = 0xA0, Default=0x00

Bit(s)	Name	Description	R/W	Reset
7: 5	SIRQ_PEND_SEL	Sirq 中断模式选择： 000: reserved 001: 上升沿 010: 下降沿 011: 上升/下降沿 100: 低电平 101: 高电平 110: reserved 111: reserved	R/W	000
4		Boost 短路 scdt 打嗝控制：	R/W	0

		1: scdt 发生时硬件每隔 1s 打嗝一次, 如果在 4s 时间内打嗝了 3 次则关闭 bst 并发出中断信号 0: scdt 发生时只发中断信号		
3		Boost 低压 Lowpot 打嗝控制: 1: lowpot 发生时硬件每隔 500ms 打嗝一次, 如果在 2s 时间内打嗝了 3 次则关闭 bst 并发出中断信号 0: lowpot 发生时只发中断信号	R/W	0
2		boost 在刚出打嗝信号的时候就唤醒和中断: 0: 不使能该功能 1: 使能该功能	R/W	0
1	--	Reserved		
0	EA	系统中断总使能: 1: enable 0: disable	RW	0

IRQ_EN0(0xA4)

Offset = 0xA4, Default=0x00

Bit(s)	Name	Description	R/W	Reset
7	Boost_SCDT_EN	Boost 短路异常 SCDT 中断使能 1: Enable 0: Disable	R/W	0
6	Boost_Lowpot_EN	Boost 低压异常 Lowpot 中断使能 1: Enable 0: Disable	R/W	0
5	BatLow_IRQ_EN	电池低电中断使能 1: Enable 0: Disable	R/W	0
4	LongKey_IRQ_EN	KEY 长按中断使能 1: Enable 0: Disable	R/W	0
3	DoubleKey_IRQ_EN	KEY 双击中断使能 1: Enable 0: Disable	R/W	0
2	ShortKey_IRQ_EN	KEY 短按中断使能 1: Enable 0: Disable	R/W	0
1	KeyUp_IRQ_EN	KEY 抬起中断使能 1: Enable 0: Disable	R/W	0
0	KeyDown_IRQ_EN	KEY 按下中断使能 1: Enable 0: Disable	R/W	0

IRQ_EN1(0xA5)

Offset = 0xA5, Default=0x00

Bit(s)	Name	Description	R/W	Reset
7	Watchdog_IRQ_EN	Watchdog 中断使能 1: Enable 0: Disable	R/W	0
6	Timer1_IRQ_EN	Timer1 中断使能 1: Enable 0: Disable	R/W	0
5	Timer0_IRQ_EN	Timer0 中断使能 1: Enable 0: Disable	R/W	0
4	—	Reserved	R/W	—
3	VINUV_IRQ_EN	VIN 欠压中断使能 1: Enable 0: Disable	R/W	0
2	VINOV_IRQ_EN	VIN 过压中断使能 1: Enable 0: Disable	R/W	0
1	VIN_OUT_IRQ_EN	VIN 拔出中断使能 1: Enable 0: Disable	R/W	0
0	VIN_IN_IRQ_EN	VIN 插入中断使能 1: Enable 0: Disable	R/W	0

IRQ_EN2(0xA6)

Offset = 0xA6, Default=0x00

Bit(s)	Name	Description	R/W	Reset
7	OT_IRQ_EN	OT 过温中断使能: 1: Enable 0: Disable	R/W	0
6:3	—	Reserved	R/W	0
2	SIRQ_EN	SIRQ 中断使能 1: Enable 0: Disable	R/W	0
1:0	—	Reserved		

5.4 中断优先级

IP5902 内核共有 4 个中断优先级，其配置如下：

ip1.x	ip0.x	Priority Level
0	0	Level0 (lowest)
0	1	Level1
1	0	Level2
1	1	Level3 (highest)

Bits	Group	
IP1.0, IP0.0	SCDT Lowpot	
IP1.1, IP0.1	Watchdog	
IP1.2, IP0.2	Timer0	KEY
IP1.3, IP0.3	Timer1	
IP1.4, IP0.4	SIRQ	
IP1.5, IP0.5	Batlow_irq	VINOV VINUV VIN_IN VIN_OUT OT

INT_PRO0(0x90)

Offset = 0x90, Default=0x00

Bit(s)	Name	Description	R/W	Reset
7	–		R/W	0
6	–		R/W	0
5	IP0. 5		R/W	0
4	IP0. 4		R/W	0
3	IP0. 3		R/W	0
2	IP0. 2		R/W	0
1	IP0. 1		R/W	0
0	IP0. 0		R/W	0

INT_PRO1(0x91)

Offset = 0x91, Default=0x00

Bit(s)	Name	Description	R/W	Reset
7	–		R/W	0
6	–		R/W	0
5	IP1. 5		R/W	0
4	IP1. 4		R/W	0
3	IP1. 3		R/W	0
2	IP1. 2		R/W	0
1	IP1. 1		R/W	0
0	IP1. 0		R/W	0

6. 系统时钟

芯片运行所需要的时钟源由振荡器提供，本芯片所提供的振荡器有两种：

12MHz 高频 RC 振荡器 HOSC；

32KHz 低频 RC 振荡器 LOSC；

CPU 的时钟源固定为系统主时钟 HOSC，可以设置 CPU 的时钟频率 F_{CLK} 固定为 HOSC 的 1 分频、3 分频、4 分频、6 分频。

6.1 高频时钟

芯片内置 1 个振荡频率为 12MHz 的高精度 HOSC 振荡器，其分频时钟用作系统主时钟源。

6.2 低频时钟

芯片内置 1 个振荡频率为 32KHz 的 LOSC 振荡器，LOSC 在工作状态和低功耗状态为相关电路提供时钟。

6.3 时钟配置相关寄存器

CMU_CTL0(0x9A)

Offset = 0x9A, Default=0xFA

Bit(s)	Name	Description	R/W	Reset
7	ADC_CLK_RST	0:reset 1:normal	R/W	1
6	TIMER_CLK_RST	0:reset 1:normal	R/W	1
5	—	Reserved	R/W	1
4	ADC_CLK_EN	ADC Clock enable 0:disable 1:enable	R/W	1
3	TIMER_CLK_EN	Timer Clock enable 0:disable 1:enable	R/W	1
2:0	—	Reserved		

7.I/O 端口

7.1 I/O 口功能

通用 IO 口支持三状态：输入、输出及高阻态；

推挽输出时，有强弱两种驱动电流（10mA/5mA）；

每个 IO 内部都有弱上拉或者弱下拉（100K）；

GPIO0 和 GPIO1 通过唤醒后检测 GPIO0 和 GPIO1 的电平来决定是否用作 ICP 功能。开机时系统将 GPIO1 下拉 100K 左右，若 GPIO1 为高电平，则 GPIO0 和 GPIO1 进入 ICP 功能；否则 GPIO0 和 GPIO1 当做正常功能使用。

MFP 复用关系：

序号	通用功能	复用功能			烧录
IO7	GPIO7				
IO6	GPIO6				
IO5	GPIO5	PWM1H			
IO4	GPIO4	PWM0	SIRQ		
IO3	GPIO3	PWM0	ADC_VGP3	SIRQ	
IO2	GPIO2	PWM1L	ADC_VGP2	SIRQ	
IO1	GPIO1	PWM1H	ADC_VGP1	SIRQ	ICP_CLK
KEY	KEY	GPIO0			ICP_DATA

表 1 IP5902 IO 口功能表

7.2 I/O 端口配置寄存器

7.2.1 I/O 端口通用配置寄存器

PAD_DRV(0xC5)

Offset = 0xC5 default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	DRV_EH	对应 GPIO7~GPIO0 的驱动能力： 0：弱 1：强	R/W	0x00

GPIO_PD(0xC6)

Offset = 0xC6 default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	GPIO_PD_EN	对应 GPIO7~GPIO0 的下拉使能 1：Enable 0：Disable	R/W	0x00

GPIO_PU(0xC7)

Offset = 0xC7 default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	GPIO_PU	对应 GPIO7~GPIO0 的上拉使能 1: Enable 0: Disable	R/W	0x00

GPIO_OE(0xC8)

Offset = 0xC8 default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	GPIO_OE	对应 GPIO7~GPIO0 的输出使能	R/W	0x00

GPIO_IE(0xC9)

Offset = 0xC9 default=0x00

Bit(s)	Name	Description	R/W	Reset
7: 0	GPIO_IE	对应 GPIO7~GPIO0 的输入使能	R/W	0x00

7.2.2 I/O 端口数据寄存器

GPIO_DAT(0xCA)

Offset = 0xCA default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	GPIO_DATA	对应 GPIO7~GPIO0 的输入或输出数据	R/W	0x00

7.2.3 I/O 端口复用配置寄存器

MFP_CTL0(0xCB)

Offset = 0xCB default=0x00

Bit(s)	Name	Description	R/W	Reset
7	GPIO7	0: GPIO7 1: -	R/W	0
6	GPIO6	0: GPIO6 1: -	R/W	0
5	GPIO5	0: GPIO5 1: PWM1H	R/W	0
4:3	GPIO4	0: GPIO4 1: PWM0 2: SIRQ 3: -	R/W	00

2:0	GPI03	0: GPI03 1: ADC_VGP3 2: PWM0 3: - 4: - 5: - 6: SIRQ 7: -	R/W	000
-----	-------	---	-----	-----

MFP_CTL1(0xCC)

Offset = 0xCC default=0x00

Bit(s)	Name	Description	R/W	Reset
7:5	GPI02	0: GPI02 1: ADC_VGP2 2: PWM1L 3: - 4: - 5: - 6: SIRQ 7: -	R/W	000
4:2	GPI01	0: GPI01 1: ADC_VGP1 2: PWM1H 3: - 4: - 5: - 6: SIRQ 7: -	R/W	000
1	GPI00	0: KEY 1: GPI00	R/W	0
0		Reserved		

8. 硬件按键 Key

8.1 Key 模块介绍

IP5902 支持专门的按键检测引脚 Key，按键的结果通过只读 bit 放到寄存器上，CPU 读到这些 bit 再处理相关动作。按键动作包括：

- 1、短按：60ms~1s 以内的低电平
 - 2、双击：在 1s 时间内出现两次短按
 - 3、长按：持续出现低电平超过 2s
 - 4、超长按：持续出现低电平超过 10s
 - 5、按下去：电平由高转低，持续 60ms 以上
 - 6、抬起来：电平由低转高，持续 60ms 以上
 - 7、上升沿+下降沿：debounce 2ms。
- 除超长按外，其他六种按键动作均有相应中断功能。

8.2 key 状态相关寄存器

DEBUG4(0x8C)

Offset = 0x8C, Default=0x29

Bit(s)	Name	Description	R/W	Reset
7:6	–	Reserved	R	–
5	KEY_STAT_R	按键状态指示： 1：未按下 0：按下	R	1
5:0	–	Reserved	R	–

9. 定时器

9.1 看门狗定时器

芯片内建一个看门狗定时器，它可以在发生软件故障时，将芯片复位。使能看门狗时，若用户程序异常、清除看门狗定时器失败，则在预定的时间到达后，看门狗会发出中断信号或者直接使系统复位。

IP5902集成了WDT定时器，特性如下：

- 1、 定时器时钟源为内部 32KHz RC 时钟
- 2、 8 档 WDT 定时计数器
- 3、 中断功能
- 4、 复位功能
- 5、 唤醒功能

WDCTL(0xC4)

Offset = 0xC4, Default=0x00

Bit(s)	Name	Description	R/W	Reset
7	WD_EN	Watchdog 使能: 0: Disable 1: Enable	R/W	0
6:4	WD_CLK	Watchdog 时间选择 (RC_LOSC source, 32K): CLK Watch Dog Length 000 2hours 001 64s 010 128 ms 011 256 ms 100 1 s 101 2 s 110 4 s 111 8 s 注意: 修改 Watchdog 的时间的时候, 需要先 Disable WD_EN 再改时间再 Enable WD_EN	R/W	000
3: 2	WD_MODE	Watchdog Signal (IRQ or Reset)选择 11: Reserved 10: 无 reset 功能, 只发出唤醒系统信号 01: reset 后进 standby, 是否再唤醒系统根据 WK_CTL0 的 bit0 来决定 00: IRQ	R/W	00
1:0	-	Reserved	R/W	-

WD_CLEAR(0x80)

Offset = 0x80, Default=0x00

Bit(s)	Name	Description	R/W	Reset
--------	------	-------------	-----	-------

7:1	—	Reserved	R/W	—
0	CLEAR	每次写 1 动作会清 watchdog 计数器。 读取的值为正常写入的值。	R/W	0

9.2 Timer0/Timer1 定时器/计数器

支持 2 个定时器 Timer0/1，且各有一个 pwm 控制器，Timer1 有互补 pwm 功能：

均支持三种计数模式：

1、Normal Mode：定时器从 0 开始累加计数，timer 计时到 Timer_Val 后自动停止。

2、Circle mode：定时器从 0 开始累加计数，计数值加到目标值后，发出中断标志，且计数器持续累加，直到累加到最大值 0xFFFF 溢出后又从 0 开始累加，加到目标值后又发出中断标志，如此不断循环。

3、Reload mode：定时器从 0 开始累加计数，计数值加到目标值后，发出中断标志，然后计数值清零，从零开始重新计数。

启动 timer 时，需先配置好 timer clk/mode/time_val/time_pwm 等，然后 enable timer。

9.2.1 Timer0 定时器/计数器相关寄存器

T0_CTL(0xBA)

Addr=0xBA default=0x00

Bit(s)	Name	Description	R/W	Reset
7	TIMERO_EN	1: enable 0: disable 写 1 启动 timer，如果是 normal 模式，计数完成后硬件自动将 enable 清 0	RW	0
6:4	TIMERO_CLK_DIV	Timer0 clock source divider: 000: HCLK/1 001: HCLK/2 010: HCLK/4 011: HCLK/16 100: HCLK/32 101: HCLK/64 110: HCLK/128 111: HCLK/256	RW	0x0
3:2	TIMERO_MODE	Timer0 Mode 00: Reserved (timer disable) 01: Circle mode 10: Normal mode 11: Reload mode	RW	00
1	—	Reserved	RW	0
0	PWMO_POLARITY	PWMO polarity control:	RW	0x0

		0: The Timer0 Value $\geq$ TO_PWM, out 1 the Timer0 Value $<$ TO_PWM, out 0 1: The Timer0 Value $\geq$ TO_PWM, out 0 the Timer0 Value $<$ TO_PWM, out 1		
--	--	--	--	--

T0_PL(0xBB)

Timer0 Period Low Register

Addr=0xBB default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	TOPL	Timer0 Period Low, when the Timer count reach (TOPH $<$ 8 + TOPL), the Timer pending=1.	R/W	0x0

T0_PH(0xBC)

TIMER0 Period High Register

Addr=0xBC default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	TOPH	Timer0 Period High, when the Timer count reach TOP, the timer pending=1	R/W	0x0

9.2.2 Timer1 定时器/计数器相关寄存器

T1_CTL(0xBF)

Timer1 Control Register

Addr=0xBF default=0x00

Bit(s)	Name	Description	R/W	Reset
7	TIMER1_EN	1: enable 0: disable 写1启动 timer, 如果是 normal 模式, 计数完成后硬件自动将 enable 清0	RW	0
6:4	TIMER1_CLK_DIV	Timer1 clock source divider: 000: HCLK/1 001: HCLK/2 010: HCLK/4 011: HCLK/16 100: HCLK/32 101: HCLK/64 110: HCLK/128 111: HCLK/256	RW	0x0
3:2	TIMER1_MODE	Timer1 Mode 00: Reserved 01: Circle mode 10: Normal mode	RW	00

		11: Reload mode		
1	PWM1_HL_CTL	1: PWM1H 是 PWM1L 的反相 0: PWM1H 是 PWM1L 的同相	RW	0
0	PWM1_POLARITY	PWM1 polarity control: 0: The Timer1 Value $\geq$ T1_PWM, out 1 the Timer1 Value $<$ T1_PWM, out 0 1: The Timer1 Value $\geq$ T1_PWM, out 0 the Timer1 Value $<$ T1_PWM, out 1	RW	0x0

T1_PL(0xC0)

Timer1 Period Low Register

Addr=0xC0 default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	T1PL	Timer1 Period Low, when the Timer count reach (T1PH $<$ 8 + T1PL), the timer pending=1 .	RW	0x0

T1_PH(0xC1)

Timer1 Period High Register

Addr=0xC1 default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	T1PH	Timer1 Period High, when the Timer count reach T1P, the Timer pending=1 .	R/W	0x0

9.3 PWM

芯片内建两个 PWM 可复用到多个 IO 上输出，支持 16bit 占空比分级。

- 1、PWM0 与 Timer0 复用。
- 2、PWM1 与 Timer1 复用。

9.3.1 PWM0 相关寄存器

T0_PWML(0xBD)

Timer0 PWM Low Register

Addr=0xBD default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	TOPWML	Timer0 PWM Low for PWM inversion, $(TOPWMH \ll 8 + TOPWML) / (TOPH \ll 8 + TOPL)$ is PWM' s duty cycle	R/W	0x0

T0_PWMH(0xBE)

Timer0 PWM High Register

Addr=0xBE default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	TOPWMH	Timer0 PWM High for PWM inversion, $(TOPWMH \ll 8 + TOPWML) / (TOPH \ll 8 + TOPL)$ is PWM' s duty cycle	R/W	0x0

9.3.2 PWM1 相关寄存器

T1_PWML(0xC2)

Timer1 PWM Low Register

Addr=0xC2 default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	T1PWML	Timer1 PWM Low for PWM inversion, $(T1PWMH \ll 8 + T1PWML) / (T1PH \ll 8 + T1PL)$ is PWM' s duty cycle	R/W	0x0

T1_PWMH(0xC3)

Timer1 PWM High Register

Addr=0xC3 default=0x00

Bit(s)	Name	Description	R/W	Reset
7:0	T1PWMH	Timer1 PWM High for PWM inversion, $(T1PWMH \ll 8 + T1PWML) / (T1PH \ll 8 + T1PL)$ is PWM' s duty cycle	R/W	0x0

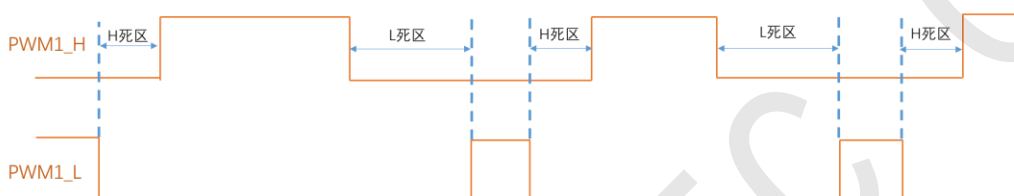
PWM1_CTL1(0x95)

Addr =0x95, Default = 0x00

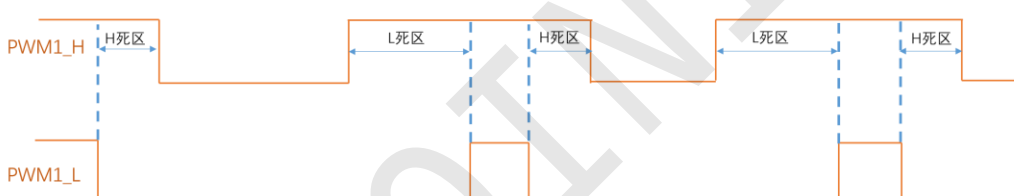
Bit(s)	Name	Description	R/W	Reset
7:4	—	PWM1 H 死区时间: HCLK 数 000000: 无死区 000001: 死区时间为 2 个 clock 000002: 死区时间为 4 个 clock 死区时间=2*N* (hclk)	RW	0000
3:0	—	PWM1 L 死区时间: HCLK 数 000000: 无死区 000001: 死区时间为 2 个 clock 000002: 死区时间为 4 个 clock 死区时间=2*N* (hclk)	RW	0000

互补 pwm 示意图:

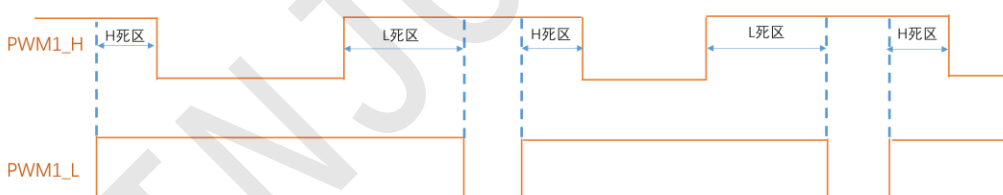
PWM1_H和PWM1_L反向, PWM1 Polarity=1,H死区和L死区配成不一样的数值



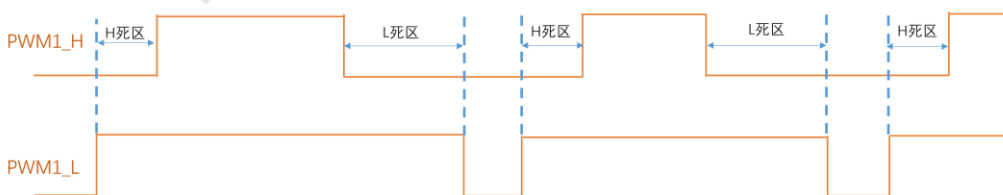
PWM1_H和PWM1_L同向, PWM1 Polarity=1, H死区和L死区配成不一样的数值



PWM1_H和PWM1_L反向, PWM1 Polarity=0,H死区和L死区配成不一样的数值



PWM1_H和PWM1_L同向, PWM1 Polarity=0,H死区和L死区配成不一样的数值



10.模数转换（ADC）

10.1 ADC 模块介绍

IP5902 集成一个 10bit ADC，用来检测 8 路输入信号。

整个 ADC 的输入时钟为 1MHz，DOUT[9:0]数据时钟为 71.4KHz（每 14 个 1MHz 时钟周期出一组 ADC 数据），8 通道全开时每隔 0.112ms 出一个 adc 数据，如果 128 次平均，则 14.3ms 出一次 adc 数据。

ADC 为 1M 的工作时钟，8 通道分别为 VIN 电压、ICRG 电流、VBAT 电压、VGP1 电压、VGP2 电压、VGP3 电压、VOUT 电压、VND 电压。

其中 ADC1/ADC2/ADC3 是 IO1/2/3 的复用功能，同时 IC 可从对应 IO 放出 20uA 或 100uA 恒流源，可以在 ADC pin 外挂电阻再通过 ADC 测量此时电阻在 20uA 下的压降从而实现 PIN 选功能，比如 IP5902 可以通过在 ADC pin 上接不同的电阻，来选择电池的恒流充电电流。

其中 ADC2 是 IO2 的复用功能，IC 可从 IO2 放出 20uA/100uA 恒流源，可以在 ADC pin 外挂电阻再通过 ADC 测量此时电阻在 20uA/100uA 下的压降，从而实现检测外部电阻阻值功能，比如 IP5902 可以通过在 ADC pin 上接 NTC 电阻，检测 NTC 电阻阻值，对比 NTC 电阻的 R-T 表得到当前 NTC 的温度。

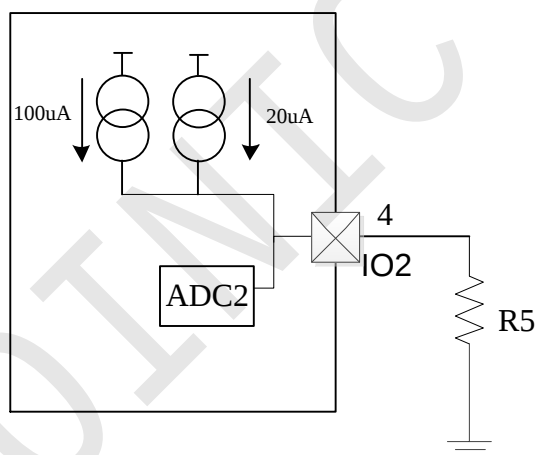


图 3 恒流充电电流设置电路

10.2 ADC 相关寄存器汇总

10.2.1 ADC 配置寄存器

ADC_ANA_CTL1(0XE7)

Offset = 0XE7, Default=0x01

Bit(s)	Name	Description	R/W	Reset
7:5	ENIPUSW<3:1>	GPIO 上拉电流切换 0: 20uA 1: 100uA Bit7: GPIO3 Bit6: GPIO2 Bit5: GPIO1	R/W	000 OTP
4:2	ENIPU_GPIO<3:1>	GPIO 上拉电流使能 <2:0>对应 GPIO3/2/1 0: disable 1: enable Bit4: GPIO3 Bit3: GPIO2 Bit2: GPIO1	R/W	000 OTP
1: 0	-	Reserved		

ADC_CTL0(0xAA)

Addr = 0xAA Default=0x07

Bit(s)	Name	Description	R/W	Reset
7	EN_VGP3	VGP3 使能信号: 0: 该路 ADC 不使能 1: 该路 ADC 使能	RW	0
6	EN_VGP2	VGP2 使能信号: 0: 该路 ADC 不使能 1: 该路 ADC 使能	RW	0
5	EN_VGP1	VGP1 ADC 使能信号: 0: 该路 ADC 不使能 1: 该路 ADC 使能	RW	0
4	EN_VND	VND ADC 使能信号: 0: 该路 ADC 不使能 1: 该路 ADC 使能	RW	0
3	EN_VOUT	VOUT ADC 使能信号: 0: 该路 ADC 不使能 1: 该路 ADC 使能	RW	0
2	EN_VBAT	VBAT ADC 使能信号: 0: 该路 ADC 不使能 1: 该路 ADC 使能	RW	1
1	EN_ICHG	ICHG ADC 使能信号: 0: 该路 ADC 不使能 1: 该路 ADC 使能	RW	1
0	EN_VIN	VIN ADC 使能信号: 0: 该路 ADC 不使能	RW	1

1: 该路 ADC 使能

ADC_CTL1(0xAB)

Addr = 0xAB Default=0x23

Bit(s)	Name	Description	R/W	Reset
7:3	–	Reserved		
2:0	ADC_AVT	ADC 数据求平均次数选择: 000: 4 次平均 001: 32 次平均 010: 64 次平均 011: 128 次平均 (14.3ms 出一组 8 通道数据) 100: 256 次平均 (28.6ms 出一组 8 通道数据) 101: 512 次平均 110: 1024 次平均 111: 2048 次平均	R/W	011

10.2.2 ADC_VIN 数据寄存器

VIN_DAT0(0xAC)

Addr = 0xAC

Bit(s)	Name	Description	R/W	Reset
7:0	VIN_H8b	LSB=17.1575mV $VIN = (ADC[9:0] + 0.5) * 17.1875 - 2800 \text{ (mV)}$ 对应量程约-2.8V~14.8V	R	0

VIN_DAT1(0xAD)

Addr = 0xAD

Bit(s)	Name	Description	R/W	Reset
7:2	–	Reserved	R	0
1:0	VIN_L2b		R	0

ADC 寄存器需要每次 read 都读两个寄存器，先读高 8bits，再读低 2bits，两次 read 过程中 ADC 寄存器不更新。

10.2.3 ADC_ICHG 数据寄存器

ICHG_DAT0(0xAE)

Addr = 0xAE

Bit(s)	Name	Description	R/W	Reset
7:0	ICHG_H8b	LSB=1.375mA $ICHG = (ADC[9:0] + 0.5) * 1.375 - 224 \text{ (mA)}$ 对应量程约-224mA~1185mA	R	0

ICHG_DAT1(0xAF)

Addr = 0xAF

Bit(s)	Name	Description	R/W	Reset
--------	------	-------------	-----	-------

7:2	-	Reserved	R	-
1:0	ICHG_L2b		R	0

ADC 寄存器需要每次 read 都读两个寄存器，先读高 8bits，再读低 2bits，两次 read 过程中 ADC 寄存器不更新。

10.2.4 ADC_VBAT 数据寄存器

VBAT_DAT0(0xB0)

Addr = 0xB0

Bit(s)	Name	Description	R/W	Reset
7:0	VBAT_H8b	LSB=4.296875mV VBAT = (ADC[9:0]+0.5) * 4.296875+400 (mV) 对应量程约 400mV~4800mV	R	0

VBAT_DAT1(0xB1)

Addr = 0xB1

Bit(s)	Name	Description	R/W	Reset
7:2	-	Reserved	R	-
1:0	VBAT_L2b		R	0

ADC 寄存器需要每次 read 都读两个寄存器，先读高 8bits，再读低 2bits，两次 read 过程中 ADC 寄存器不更新。

10.2.5 ADC_VGP1 数据寄存器

VGP1_DAT0(0xB6)

Addr = 0xB6

Bit(s)	Name	Description	R/W	Reset
7:0	ADC_VGP1_H8b	LSB=4.296875mV VGP1=(ADC[9:0]+0.5) * 4.296875-700(mV)	R	0

VGP1_DAT1(0xB7)

Addr = 0xB7

Bit(s)	Name	Description	R/W	Reset
7:2	-	Reserved	R	-
1:0	ADC_VGP1_L2b		R	0

ADC_VGP1 对应 IO1

ADC 寄存器需要每次 read 都读两个寄存器，先读高 8bits，再读低 2bits，两次 read 过程中 ADC 寄存器不更新。

10.2.6 ADC_VGP2 数据寄存器

VGP2_DAT0(0xB8)

Addr = 0xB8

Bit(s)	Name	Description	R/W	Reset
7:0	ADC_VGP2_H8b	LSB=4.296875mV $VGP2=(ADC[9:0]+0.5) * 4.296875-700(mV)$	R	0

VGP2_DAT1(0xB9)

Addr = 0xB9

Bit(s)	Name	Description	R/W	Reset
7:2	-	Reserved	R	-
1:0	ADC_VGP2_L2b		R	0

ADC_VGP2 对应 IO2

ADC 寄存器需要每次 read 都读两个寄存器，先读高 8bits，再读低 2bits，两次 read 过程中 ADC 寄存器不更新。

10.2.7 ADC_VGP3 数据寄存器

VGP3_DAT0(0xE3)

Addr = 0xE3

Bit(s)	Name	Description	R/W	Reset
7:0	ADC_VGP3_H8b	LSB=4.296875mV $VGP3=(ADC[9:0]+0.5) * 4.296875-700(mV)$	R	0

VGP3_DAT1(0xF6)

Addr = 0xF6

Bit(s)	Name	Description	R/W	Reset
7:2	-	Reserved	R	-
1:0	ADC_VGP3_L2b		R	0

ADC_VGP3 对应 IO3

ADC 寄存器需要每次 read 都读两个寄存器，先读高 8bits，再读低 2bits，两次 read 过程中 ADC 寄存器不更新

10.2.8 ADC_VOUT 数据寄存器

VOUT_DAT0(0xB2)

Addr = 0xB2

Bit(s)	Name	Description	R/W	Reset
7:0	VOUT_ADC_H8b	LSB=17.1575mV $VOUT=(ADC[9:0]+0.5) * 17.1875-2800 (mV)$ 对应量程约-2.8V~14.8V	R	0

VOUT_DAT1(0xB3)

Addr = 0xB3

Bit(s)	Name	Description	R/W	Reset
7:2	-	Reserved	R	-
1:0	VOUT_ADC_L2b		R	0

ADC 寄存器需要每次 read 都读两个寄存器，先读高 8bits，再读低 2bits，两次 read 过程中 ADC 寄存器不更新。

10.2.9 ADC_VND 数据寄存器

VND_DAT0(0xB4)

Addr = 0xB4

Bit(s)	Name	Description	R/W	Reset
7:0	VND_ADC_H8b	LSB=0.5371mV VND=(ADC[9:0]+0.5) *0.5371-275(mV)	R	0

VND_DAT1(0xB5)

Addr = 0xB5

Bit(s)	Name	Description	R/W	Reset
7:2	-	Reserved	R	-
1:0	VND_ADC_L2b		R	0

ADC 寄存器需要每次 read 都读两个寄存器，先读高 8bits，再读低 2bits，两次 read 过程中 ADC 寄存器不更新。

11.Charger 模块

11.1 模块介绍

IP5902集成一个线性充电charger，最大充电电流0.8A，充电电流大小25mA/档寄存器可调节。

11.2Charger 相关寄存器

CHG_CTL0(0x9C)

Offset = 0x9C, Default=0x00

bit	Name	Description	R/W	Reset
7	-	Reserved	R/W	0
6:2	CHG_ISET	$I_{cc}=(1+iset)*25mA$ 充电电流值配置 00000: 25mA ... 11111: 800mA	R/W	0
1:0	CHG_EN	00: 不使能充电 01: Reserved 10: Reserved 11: 使能充电	R/W	00

CHG_STATE0(0x9D)

Offset = 0x9D, Default=0x00

bit	Name	Description	R/W	Reset
7	-	Reserved	R	-
6	CHG_TK_R	涓流 0: 非涓流状态 1: 涓流状态	R	0
5	-	Reserved	R	-
4	CHG_END_R	充满 0: 没充满 1: 充满	R	0
3:0	-	Reserved	R	0

TOP_CTL2(0xEE)

Offset = 0xEE, Default=0x1F

bit	Name	Description	R/W	Reset
7:6	RVINOV<1:0>	VINOV 阈值 11: 15V 10: 10.5V 01: 6.5V 00: 5.6V	RW	00
5	RVINUV	VINUV 阈值 1: 4.6V 0: 4.25V	RW	0
4	ENBATLOW	使能低电检测 1: 使能 0: 不使能	RW	1
3:2	BATLOWSET _STD<1:0>	电池低电关机电压设置 00: 2.7 01: 2.8 10: 2.9 11: 3	RW	11
1:0	BATLOWSET _IRQ<1:0>	电池低电中断电压设置 00: 2.8 01: 2.9 10: 3 11: 3.1	RW	11

CHG_ANA_CTL1(0xEB)

Offset = 0xEB, Default=0x1F

bit	Name	Description	R/W	Reset
7:6	RVCHG<1:0>	电池充满电压设置 00: 4.2V 01: 4.3V 10: 4.35V 11: 4.4V	RW	00
5:2	-	Reserved		
1:0	RVIL	充电 VIN 欠压环设置 11: 4.7V 10: 4.6V 01: 4.5V 00: 4.4V	RW	11

12.Boost 模块

12.1 Boost 模块介绍

IP5902 集成一个异步 Boost 升压控制器，输出电压 4V~9V 可调，最大输出功率 9V/1A。内置软启动功能，防止在启动时的冲击电流过大引起故障，集成输出过流，短路，过压，过温等保护功能，确保系统稳定可靠的工作。

12.2 Boost 相关寄存器

12.2.1 Boost 状态相关寄存器

DEBUG1(0x89)

Offset = 0x89, Default=0x1C

Bit(s)	Name	Description	R/W	Reset
7	BSTPWROK_R	BOOST 输出电压正常标志	R	0
6	NPL_R	输出过压标志	R	0
5	-	Reserved		
4	BATPOR_R	BAT 电压高于 2.4V 为 1	R	1
3	VINOK_R	VIN 电压有效标志	R	1
2:0	-	Reserved		

BST_SET(0xA7)

Addr = 0xA7 Default = 0x00

bit	Name	Description	R/W	Reset
7	BOOST_EN	BOOST 升压输出使能 1: Enable 0: Disable	R/W	0
6	-	Reserved		
5: 0	BST_VSET<5:0>	BST VOUT 输出电压挡位,step=0.2V 00 0000: 4.0V 00 0001: 4.2V ... 00 0101: 5.0V ... 00 1010: 6.0V ... 00 1111: 7.0V ... 01 0100: 8.0V ... 01 1001: 9.0V	R/W	00 0000

		超过 011001 的，不能使用		
--	--	------------------	--	--

13.VND 路径 MOS 模块

13.1 VND 路径 MOS 介绍

IP5902 集成一个路径 NMOS，路径 NMOS 的 S 端接 GND，D 端为 VND 引脚；内部的路径 NMOS 的导通阻抗 $R_{ds(on)}=130$ 毫欧，最大电流不超过 2A；

13.2 VND 路径 MOS 相关寄存器

12.2.1 VND 路径 MOS 控制寄存器

VND_CTL(0x96)

Offset = 0x96, Default=0x2B

Bit(s)	Name	Description	R/W	Reset
7	VND_EN	内部路径 MOS 使能 0: 不使能，关闭内部路径 MOS 1: 使能，打开内部路径 MOS	RW	0
6:0	-	Reserved	RW	010 1011

14.典型应用原理图

IP5902 只需要电感、电容、电阻，即可实现完整功能的充电方案。

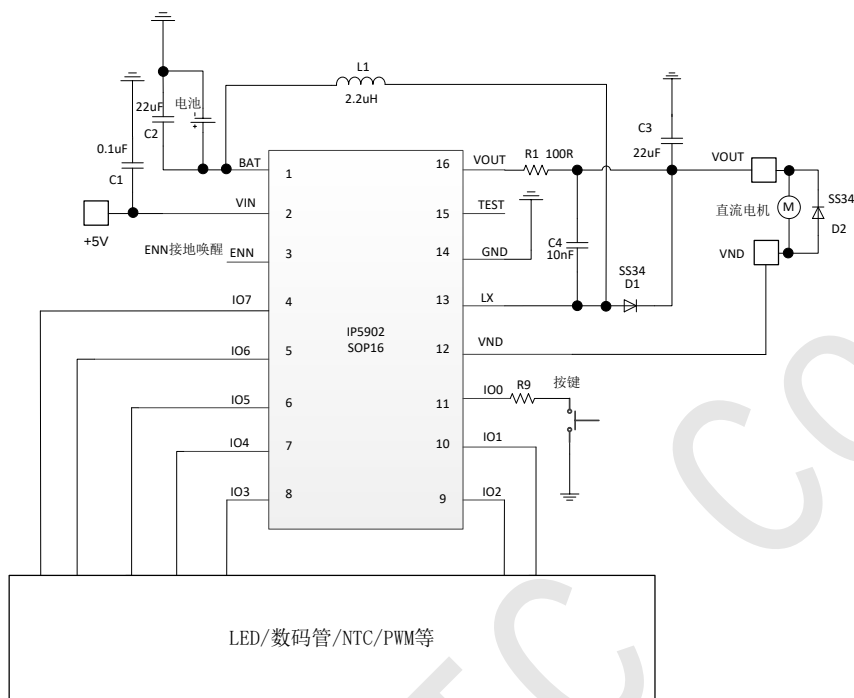
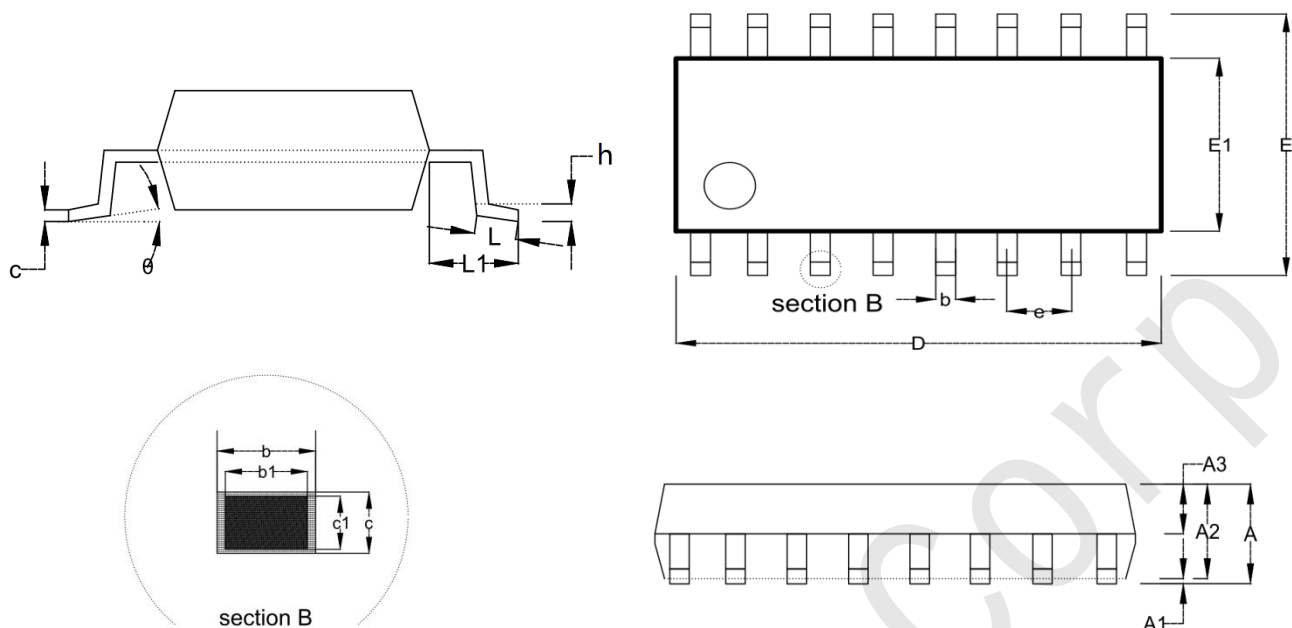


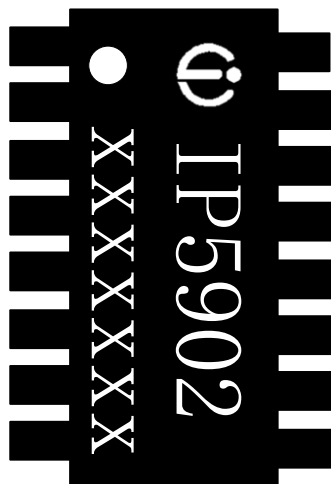
图 7 IP5902 原理图

14.封装信息



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	--	--	1.75
A1	0.05	--	0.225
A2	1.30	1.40	1.50
A3	0.60	0.65	0.70
b	0.39	--	0.48
b1	0.38	0.41	0.43
c	0.21	--	0.26
c1	0.19	0.20	0.21
D	9.70	9.90	10.10
E	5.80	6.00	6.20
E1	3.70	3.90	4.10
e	1.27BSC		
h	0.25	--	0.50
L	0.50	--	0.80
L1	1.05BSC		
θ	0	--	8°

15.IC 丝印说明



说明:

- 1、IP5902 --产品型号
- 2、 --英集芯标志
- 3、XXXXXXX --生产批号
- 4、○ --PIN1脚的位置标识

图 8 IP5902 IC 丝印说明

16. 责任及版权声明

英集芯科技有限公司有权对所提供的产品和服务进行更正、修改、增强、改进或其它更改，客户在下订单前应获取最新的相关信息，并验证这些信息是否完整且是最新的。所有产品的销售都遵循在订单确认时所提供的销售条款与条件。

英集芯科技有限公司对应用帮助或客户产品设计不承担任何义务。客户应对其使用英集芯的产品和应用自行负责。为尽量减小与客户产品和应用相关的风险，客户应提供充分的设计与操作安全验证。

客户认可并同意，尽管任何应用相关信息或支持仍可能由英集芯提供，但他们将独力负责满足与其产品及在其应用中使用英集芯产品相关的所有法律、法规和安全相关要求。客户声明并同意，他们具备制定与实施安全措施所需的全部专业技术和知识，可预见故障的危险后果、监测故障及其后果、降低有可能造成人身伤害的故障的发生机率并采取适当的补救措施。客户将全额赔偿因在此类关键应用中使用任何英集芯产品而对英集芯及其代理造成的任何损失。

对于英集芯的产品手册或数据表，仅在没有对内容进行任何篡改且带有相关授权、条件、限制和声明的情况下才允许进行复制。英集芯对此类篡改过的文件不承担任何责任或义务。复制第三方的信息可能需要服从额外的限制条件。

英集芯会不定期更新本文档内容，产品实际参数可能因型号或者其他事项不同有所差异，本文档不作为任何明示或暗示的担保或授权。

在转售英集芯产品时，如果对该产品参数的陈述与英集芯标明的参数相比存在差异或虚假成分，则会失去相关英集芯产品的所有明示或暗示授权，且这是不正当的、欺诈性商业行为。英集芯对任何此类虚假陈述均不承担任何责任或义务。