

LOONGSON

龙芯 1D100BA 处理器用户手册
V1.0

2026 年 7 月

龙芯中科技股份有限公司

自主决定命运, 创新成就未来

北京市海淀区中关村环保科技示范园龙芯产业园 100095
Loongson Industrial Park, Zhongguancun Environmental Protection Park,
Haidian District, Beijing 100095. P.R.China



www.loongson.cn

阅读指南

《龙芯 1D100BA 处理器用户手册》主要介绍龙芯 1D100BA 的架构与寄存器描述；包括用户手册和片上设备使用指南两部分，软件编程指南介绍软件开发过程中的常见问题。

目 录

目录	i
第一章 概述	1
1.1 特性	1
1.2 结构框图	3
1.3 文档约定	4
1.3.1 信号命名	4
1.3.2 信号类型	4
1.3.3 数值表示	4
1.3.4 寄存器域	4
第二章 地址空间	5
第三章 电源与测量控制	7
3.1 测量过程介绍	7
3.2 电源管理	7
3.3 巡航模式	8
3.4 寄存器定义	9
3.4.1 时序控制 (Timing)	10
3.4.2 测量命令、配置 (Command)	12
3.4.3 时间计数器 (Count)	13
3.4.4 唤醒时间配置 (Compare)	14
3.4.5 芯片全局配置 (ChipCtrl)	14
3.4.6 自动计算配置 (AutoCalc)	16
3.4.7 ADC 配置 (AdcCtrl)	16
3.4.8 ADC 数据 (AdcData)	17
3.4.9 外部中断使能 (ExintEn)	17
3.4.10 外部中断极性 (ExintPol)	18
3.4.11 外部中断边沿 (ExintEdge)	18
3.4.12 外部中断状态 (ExintSrc)	18
3.4.13 看门狗配置寄存器 (WdtCfg)	18

3.4.14	看门狗重置寄存器 (WdtFeed)	19
3.4.15	电源配置 (PowerCfg)	19
3.4.16	命令写端口 (CommandW)	19
3.4.17	GPIOA 输出使能 (GPIOA_OE)	20
3.4.18	GPIOA 输出电平 (GPIOA_O)	20
3.4.19	GPIOA 输入电平 (GPIOA_I)	21
3.4.20	GPIOB 输出使能 (GPIOB_OE)	21
3.4.21	GPIOB 输出电平 (GPIOB_O)	21
3.4.22	GPIOB 输入电平 (GPIOB_I)	21
3.4.23	脉冲输出配置 (Pulse0/1)	22
3.4.24	用户数据 (UserDat)	22
3.4.25	GPIO 位访问端口 (GPIOBit)	22
第四章	时差测量	23
4.1	时差测量流程	23
4.2	时差测量寄存器	23
4.2.1	发送脉冲定义 (PulseDef)	25
4.2.2	时差测量控制 (SonarCtrl)	25
4.2.3	TDC 控制 1 (TdcCtrl1)	26
4.2.4	TDC 控制 2 (TdcCtrl2)	28
4.2.5	零点校准控制 (CalibCtrl)	28
4.2.6	首波控制 (FWaveCtrl)	29
4.2.7	模拟前端控制 (AfeCtrl)	30
4.2.8	窗口延迟链配置 (DLineCtrl)	32
4.2.9	窗口配置 (GateDef)	33
4.2.10	结果指针 (ResPtr)	33
4.3	测量结果处理	34
4.3.1	结果存储	34
4.3.2	结果计算	34
4.3.3	时差测量自动计算	35
4.3.4	校准测量	36
4.3.5	高频晶振校准	36

4.4	脉冲发生器	36
4.5	窗口与波峰测量	36
4.6	测量功能拓展	37
4.7	TDC 配置与输出	38
第五章	温度测量	39
5.1	温度测量流程	39
5.2	温度测量寄存器	40
5.2.1	温度测量控制 (ThsensCtrl)	40
第六章	段式 LCD 控制器	41
6.1	概述	41
6.2	寄存器定义	41
6.2.1	帧缓冲 0 寄存器 (FB0)	41
6.2.2	帧缓冲 1 寄存器 (FB1)	41
6.2.3	帧缓冲 2 寄存器 (FB2)	42
6.2.4	刷新率寄存器 (RFR)	42
6.3	使用说明	42
6.3.1	帧缓冲排布	42
6.3.2	刷新率设置	43
第七章	中断	45
7.1	中断结构	45
7.2	中断处理	46
7.3	寄存器定义	46
7.3.1	中断使能寄存器 (INT_EN)	47
7.3.2	中断边沿寄存器 (INT_EDGE)	47
7.3.3	中断极性寄存器 (INT_POL)	48
7.3.4	中断清除寄存器 (INT_CLR)	48
7.3.5	中断置位寄存器 (INT_SET)	48
7.3.6	中断输出寄存器 (INT_OUT)	48
7.3.7	运行状态及保护寄存器 (SRPROT)	49

第八章	Flash	51
8.1	概述	51
8.2	存储空间	51
8.3	寄存器空间	51
8.3.1	命令寄存器 (CMD)	52
8.3.2	状态寄存器 (STS)	52
8.3.3	擦写时间寄存器 (PET)	52
8.3.4	BOUND 寄存器 (BOUND)	53
8.3.5	OTP 配置寄存器 (OTPCFG)	53
8.4	使用说明	53
8.4.1	代码保护	53
8.4.2	中断	54
8.4.3	编程指南	54
8.4.4	OTP 功能	54
第九章	电压测量	55
9.1	概述	55
9.1.1	模拟前端电压测量	55
9.2	寄存器定义	56
9.3	使用说明	56
第十章	I ² C 控制器	57
10.1	概述	57
10.2	主要特性	57
10.3	功能描述	58
10.3.1	模式选择	58
10.3.2	从模式	58
10.3.3	主模式	60
10.3.4	错误条件	64
10.4	中断	64
10.5	寄存器定义	65
10.5.1	I ² C 控制寄存器 1 (I ² C_CR1)	65
10.5.2	I ² C 控制寄存器 2 (I ² C_CR2)	66

10.5.3	I ² C 从地址寄存器 (I ² C_OAR)	67
10.5.4	I ² C 数据寄存器 (I ² C_DR)	67
10.5.5	I ² C 状态寄存器 1 (I ² C_SR1)	68
10.5.6	I ² C 状态寄存器 2 (I ² C_SR2)	69
10.5.7	I ² C 时钟控制寄存器 (I ² C_CCR)	69
10.5.8	I ² C 上升时间寄存器 (I ² C_TRISE)	70
第十一章 SPI 控制器		71
11.1	概述	71
11.2	寄存器定义	71
11.2.1	控制寄存器 (SPCR)	71
11.2.2	状态寄存器 (SPSR)	72
11.2.3	数据寄存器 (DATA)	72
11.2.4	外部寄存器 (SPER)	73
11.2.5	参数控制寄存器 (PARAM)	73
11.2.6	片选控制寄存器 (SOFTCS)	74
11.2.7	时序控制寄存器 (TIMING)	74
11.3	接口时序	74
11.3.1	SPI 主控制器接口时序	74
11.3.2	SPI Flash 访问时序	75
11.4	使用指南	75
11.4.1	SPI 主控制器的读写操作	75
11.4.2	硬件 SPI Flash 读	76
11.4.3	混合访问 SPI Flash 和 SPI 主控制器	77
第十二章 UART 控制器		79
12.1	概述	79
12.2	寄存器定义	79
12.2.1	数据寄存器 (DAT)	79
12.2.2	中断使能寄存器 (IER)	80
12.2.3	中断状态寄存器 (IIR)	80
12.2.4	FIFO 控制寄存器 (FCR)	80
12.2.5	线路控制寄存器 (LCR)	81

12.2.6	MODEM 控制寄存器 (MCR)	82
12.2.7	线路状态寄存器 (LSR)	82
12.2.8	MODEM 状态寄存器 (MSR)	83
12.2.9	分频值低字节寄存器 (DL_L)	83
12.2.10	分频值高字节寄存器 (DL_H)	83
12.2.11	分频值小数寄存器 (DL_D)	83
12.2.12	采样点配置寄存器 (SPC)	84
12.3	使用指南	84
12.3.1	配置流程	84
第十三章	MBUS 控制器	87
13.1	概述	87
13.2	帧过滤	87
13.3	寄存器定义	88
13.3.1	数据寄存器 (DAT)	89
13.3.2	中断使能寄存器 (IER)	89
13.3.3	中断状态寄存器 (IIR)	90
13.3.4	FIFO 控制寄存器 (FCR)	90
13.3.5	线路控制寄存器 (LCR)	91
13.3.6	bit 窗口划分和采样控制寄存器 (sample_ctrl)	91
13.3.7	线路状态寄存器 (LSR)	92
13.3.8	分频值低字节寄存器 (DL_L)	92
13.3.9	分频值高字节寄存器 (DL_H)	92
13.3.10	分频值小数寄存器 (DL_D)	93
13.3.11	发送队列中待发送的数据量 (TF_CNT)	93
13.3.12	状态寄存器 (STATUS)	93
13.3.13	帧过滤控制寄存器 0 (FSM_CTRL 0)	94
13.3.14	帧过滤前导字符值 (PRE_CHAR)	95
13.3.15	帧过滤控制寄存器 1 (FSM_CTRL 1)	95
13.3.16	帧过滤广播地址掩码 (BR_MASK)	96
13.3.17	帧过滤掩码低 8 位 (FSM_MASKL)	96
13.3.18	帧过滤掩码高 8 位 (FSM_MASKH)	96

13.3.19	帧过滤广播地址参数 (BR_CTRL)	96
13.3.20	帧过滤缓冲区中的有效数据个数 (BUF_COUNT)	97
13.3.21	帧过滤比较字节表 (FSM_TABLE)	97
13.3.22	帧过滤广播地址表 (BR_TABLE)	98
13.3.23	帧超时计数比较值的低 8 位 (FTIMEOUTL)	98
13.3.24	帧超时计数比较值的高 8 位 (FTIMEOUTH)	98
13.3.25	帧过滤环形缓冲区的读指针 (BUF_RPTR)	98
13.3.26	帧过滤环形缓冲区数据直接读取空间 (FSM_BUF)	99
13.4	配置流程	99
13.4.1	典型例子	99
13.4.2	特例	101
第十四章	实时时钟	103
14.1	概述	103
14.2	寄存器定义	103
14.2.1	分频值寄存器 (FREQ)	103
14.2.2	配置寄存器 (CFG)	103
14.2.3	时间值寄存器 0 (RTC0)	104
14.2.4	时间值寄存器 1 (RTC1)	104
14.3	说明	105
第十五章	定时器	107
15.1	概述	107
15.2	寄存器空间	107
15.2.1	配置寄存器 (CFG)	107
15.2.2	计数值寄存器 (CNT)	108
15.2.3	比较值寄存器 (CMP)	108
15.2.4	步进值寄存器 (STP)	108
15.3	使用说明	108

此页留空

表 目 录

1.1	信号类型约定	4
2.1	地址空间分布	5
2.2	小地址空间分布	5
3.1	PMU 寄存器列表	9
3.2	时序控制	10
3.3	测量命令、配置	12
3.4	时间计数器	13
3.5	唤醒时间配置	14
3.6	芯片全局配置	14
3.7	自动计算配置	16
3.8	ADC 配置	17
3.9	ADC 数据	17
3.10	外部中断使能	18
3.11	外部中断极性	18
3.12	外部中断边沿	18
3.13	外部中断状态	18
3.14	看门狗配置寄存器	19
3.15	看门狗重置寄存器	19
3.16	电源配置	19
3.17	命令写端口	20
3.18	GPIOA 输出使能	20
3.19	GPIOA 输出电平	20
3.20	GPIOA 输入电平	21
3.21	GPIOB 输出使能	21
3.22	GPIOB 输出电平	21
3.23	GPIOB 输入电平	21

3.24	脉冲输出配置	22
3.25	用户数据	22
3.26	GPIO 位访问端口	22
4.1	SONAR 寄存器列表	24
4.2	发送脉冲定义	25
4.3	时差测量控制	25
4.4	TDC 控制 1	26
4.5	TDC 控制 2	28
4.6	零点校准控制	29
4.7	首波控制	29
4.8	模拟前端控制	30
4.9	窗口延迟链配置	32
4.10	窗口配置	33
4.11	结果指针	33
4.12	测量结果存储	34
4.13	状态区测量结果排列	34
5.1	温度测量控制	40
6.1	段式 LCD 控制器寄存器列表	41
6.2	帧缓冲 0 寄存器	41
6.3	帧缓冲 1 寄存器	41
6.4	帧缓冲 2 寄存器	42
6.5	刷新率寄存器	42
6.6	帧缓冲排布表	42
7.1	中断对应关系	46
7.2	Conf 寄存器列表	47
7.3	中断使能寄存器	47
7.4	中断边沿寄存器	47
7.5	中断极性寄存器	48
7.6	中断清除寄存器	48

7.7	中断置位寄存器	48
7.8	中断输出寄存器	48
7.9	运行状态及保护寄存器	49
8.1	Flash 控制器寄存器列表	52
8.2	命令寄存器	52
8.3	状态寄存器	52
8.4	擦写时间寄存器	53
8.5	BOUND 寄存器	53
8.6	OTP 配置寄存器	53
9.1	ADC 测量通道	55
10.1	I ² C 中断请求	64
10.2	I ² C 控制器寄存器列表	65
10.3	I ² C 控制寄存器 1	65
10.4	I ² C 控制寄存器 2	67
10.5	I ² C 从地址寄存器	67
10.6	I ² C 数据寄存器	67
10.7	I ² C 状态寄存器 1	68
10.8	I ² C 状态寄存器 2	69
10.9	I ² C 时钟控制寄存器	70
10.10	I ² C 上升时间寄存器	70
11.1	SPI 控制器寄存器列表	71
11.2	控制寄存器	71
11.3	状态寄存器	72
11.4	数据寄存器	72
11.5	外部寄存器	73
11.6	SPI 分频系数	73
11.7	参数控制寄存器	73
11.8	片选控制寄存器	74
11.9	时序控制寄存器	74

12.1	UART 寄存器列表	79
12.2	数据寄存器	79
12.3	中断使能寄存器	80
12.4	中断状态寄存器	80
12.5	中断控制器功能表	80
12.6	FIFO 控制寄存器	81
12.7	线路控制寄存器	81
12.8	MODEM 控制寄存器	82
12.9	线路状态寄存器	82
12.10	MODEM 状态寄存器	83
12.11	分频值低字节寄存器	83
12.12	分频值高字节寄存器	83
12.13	分频值小数寄存器	84
12.14	采样点配置寄存器	84
13.1	CJ/T188-2004 标准帧结构	87
13.2	MBUS 寄存器列表	88
13.3	数据寄存器	89
13.4	中断使能寄存器	89
13.5	中断状态寄存器	90
13.6	中断控制器功能表	90
13.7	FIFO 控制寄存器	90
13.8	线路控制寄存器	91
13.9	bit 窗口划分和采样控制寄存器	91
13.10	线路状态寄存器	92
13.11	分频值低字节寄存器	92
13.12	分频值高字节寄存器	93
13.13	分频值小数寄存器	93
13.14	发送队列中待发送的数据量	93
13.15	状态寄存器	94
13.16	帧过滤控制寄存器 0	94
13.17	帧过滤工作模式	95

13.18 帧过滤前导字符值	95
13.19 帧过滤控制寄存器 1	95
13.20 帧过滤广播地址掩码	96
13.21 帧过滤掩码低 8 位	96
13.22 帧过滤掩码高 8 位	96
13.23 帧过滤广播地址参数	97
13.24 帧过滤缓冲区中的有效数据个数	97
13.25 帧过滤比较字节表	97
13.26 帧过滤广播地址表	98
13.27 帧超时计数比较值的低 8 位	98
13.28 帧超时计数比较值的高 8 位	98
13.29 帧过滤环形缓冲区的读指针	98
13.30 帧过滤环形缓冲区数据直接读取空间	99
14.1 实时时钟寄存器列表	103
14.2 分频值寄存器	103
14.3 配置寄存器	104
14.4 时间值寄存器 0	104
14.5 时间值寄存器 1	105
15.1 HPET 控制器寄存器列表	107
15.2 配置寄存器	107
15.3 计数值寄存器	108
15.4 比较值寄存器	108
15.5 步进值寄存器	108

此页留空

图 目 录

1.1	龙芯 1DI00BA 结构图	3
1.2	龙芯 1DI00BA QFP 封装器件标志图	3
3.1	电源架构	8
4.1	时差测量过程	23
4.2	首波模式时差测量	24
4.3	测量过程示例	35
4.4	波峰测量过程	37
4.5	外置放大器示意图	37
4.6	测量控制信号	38
5.1	温度测量充放电周期	39
5.2	一次温度测量流程	39
5.3	温度测量电路连接	39
6.1	段式 LCD 工作原理图	43
7.1	中断连接示意	45
10.1	从发送模式示意图	59
10.2	从接收模式示意图	59
10.3	主发送模式示意图	61
10.4	主接收模式（及时）示意图	63
10.5	主接收模式（非及时、3 字节）示意图	63
10.6	主接收模式（非及时、2 字节）示意图	63
11.1	SPI 主控制器接口时序	75
11.2	SPI Flash 标准读时序	75
11.3	SPI Flash 快速读时序	75
11.4	SPI Flash 双向 I/O 读时序	76

此页留空

第一章 概述

龙芯 1D100BA 是超声波热表、水表和气表测量专用 MCU 芯片。该芯片集成 CPU、FLASH、TDC、超声波脉冲发生器、温度测量单元、SPI、UART、I²C、RTC、ADC、段式 LCD 控制器等功能模块。配合完善的编程、仿真、调试环境，将会大大简化超声波流量测量系统的软硬件设计。

1.1 特性

龙芯 1D100BA 具有以下关键特性：

- 超声波时差测量
 - 测量分辨率：15ps
 - 测量范围：10us ~ 4ms
 - 回波采样数：1 ~ 32
 - 超声脉冲数：1 ~ 127
 - 超声脉冲频率：8KHz ~ 4MHz
- 温度测量
 - 2 或 4 个传感器
 - 集成施密特触发器
- 状态检测
 - 空管检测
 - 换能器断线检测
 - 电池电压检测
 - 掉电检测
- LA132 处理器核
 - 32 位单发射
 - 顺序执行、4 级流水
 - 无 cache、无 TLB
 - JTAG 调试接口支持断点、单步
 - 2 个 4KB 常开 SRAM
 - 主频 8MHz、11MHz，RAM 取指执行可达 32MHz
 - 单精度浮点协处理器，4 个浮点通用寄存器
- 片上 FLASH

- 128KB 容量
 - 每页 128 字节
- SPI 控制器
 - 2 个片选
 - 支持 SPI-Flash 启动
- UART 控制器
 - 2 路两线串口
 - 串口 0 支持红外收发
 - 串口 1 支持 CJ/TI88-2004 帧过滤，可常开
- I²C 控制器
 - 1 路
 - 支持主从模式
 - 速率 100/400Kbps
- 段式 LCD 控制器
 - 4 COM * 24 SEG
 - 1/3 偏压
- ADC
 - 2 路输入
 - 12 位分辨率，9 位有效精度
 - 最高采样率 100KHz
- 看门狗
 - 触发时间可配置，上电默认打开
 - 调试模式下暂停
- 定时器
 - 1 个 HPET
 - 支持单次、周期中断
 - 调试模式下暂停
- GPIO
 - 1 路按键输入，支持休眠唤醒
 - 最多 59 路 GPIO

1.2 结构框图

龙芯 1D100BA 由测量模块、处理器子系统、电源管理模块、通信接口等部分组成。芯片的结构如图1.1所示。

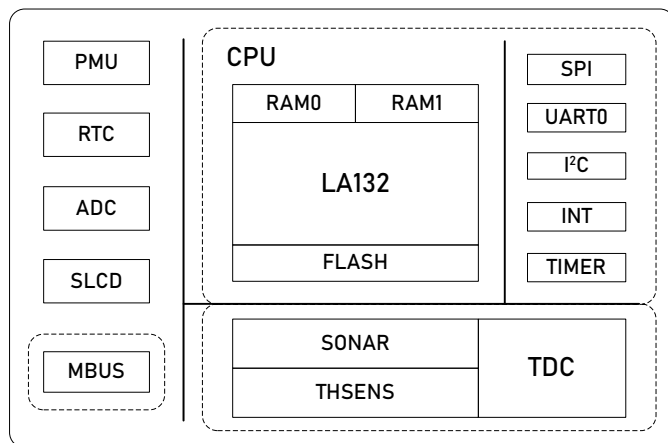


图 1.1: 龙芯 1D100BA 结构图

芯片的器件标志如图1.2所示，其中 YYWW 为器件批号，BAA 为芯片版本号。

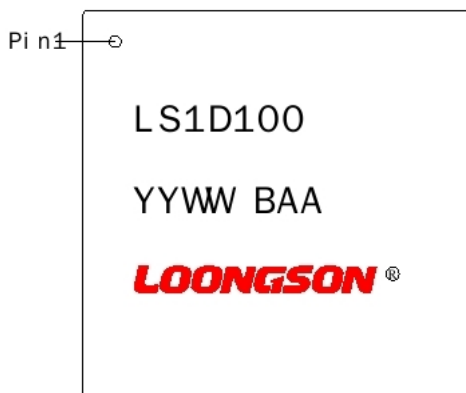


图 1.2: 龙芯 1D100BA QFP 封装器件标志图

1.3 文档约定

1.3.1 信号命名

信号名的选取以方便记忆和明确标识功能为原则。低有效信号以 $\bar{n}$ 结尾，高有效信号则不带 $\bar{n}$ 。

1.3.2 信号类型

表 1.1: 信号类型约定

代码	描述
A	模拟
I	输入
O	输出
I/O	双向
P	电源
G	地

1.3.3 数值表示

16 进制数表示为 'hxxx'，2 进制数表示为 'bxx'，其他数字为 10 进制数。

1.3.4 寄存器域

寄存器域以 [寄存器名].[域名] 的形式加以引用。如 CHIPCTRL.FastEn 指芯片配置寄存器 CHIPCTRL 的 FastEn 域。

第二章 地址空间

表 2.1: 地址空间分布

地址空间	模块	说明	访问
0x1000_0000 - 0x1000_0fff	RAM0	4KB, 可取指	BHW
0x1000_1000 - 0x1000_1fff	RAM1	4KB, 可取指	BHW
0x1c00_0000	Boot	SPI Flash or On-chip Flash	BHW
0x1e00_0000 - 0x1eff_ffff	SPI	SPI Flash	BHW
0x1f00_0000 - 0x1f01_ffff	FLASH	On-chip Flash	BHW
0x1fe6_0000 - 0x1fe6_0038	FLASH_REG	Flash regs	W
0x1fe7_0000 - 0x1fe7_0006	SPI_REG	SPI regs	B
0x1feb_0000 - 0x1feb_00bf	PMU		BW
0x1feb_4000 - 0x1feb_402c	SONAR		W
0x1feb_4080 - 0x1feb_4083	THSENS		W
0x1fe8_8000 - 0x1fe8_803f	MBUS	MBUS or UART1	B
0x1fea_0000 - 0x1fea_0006	INT		B
0x1feb_8000 - 0x1feb_800c	RTC		W
0x1feb_c000 - 0x1feb_c00c	LCD		W
0x1fed_0000 - 0x1fed_000c	TIMER		W
0x1fe9_0000 - 0x1fe9_0020	I ² C		W
0x1fe8_0000 - 0x1fe8_0006	UART0		B

表 2.2: 小地址空间分布

地址空间	模块	说明	访问
0xffff_f800 - 0xffff_ffff	RAM1 高 2KB	0x10001800 - 0x10001fff	BHW
0x0000_0500 - 0x0000_0538	FLASH_REG	Flash regs	W
0x0000_0440 - 0x0000_0446	SPI_REG	SPI regs	B
0x0000_0100 - 0x0000_01bf	PMU		BW
0x0000_0200 - 0x0000_022c	SONAR		W
0x0000_0280 - 0x0000_0283	THSENS		W
0x0000_0300 - 0x0000_033f	MBUS	MBUS or UART1	B
0x0000_0400 - 0x0000_0406	INT		B
0x0000_0410 - 0x0000_041c	RTC		W
0x0000_0420 - 0x0000_042c	LCD		W
0x0000_0430 - 0x0000_043c	TIMER		W
0x0000_0480 - 0x0000_04a0	I ² C		W
0x0000_0460 - 0x0000_0466	UART0		B

注：访问类型包括字节（B）、半字（H）和字（W），访存地址必须对齐；小地址空间模式常开，无需配置。

第三章 电源与测量控制

3.1 测量过程介绍

龙芯 1D100BA 集成了完善的时差和温度测量功能。一个测量过程包括以下几个步骤：

1. 测量事件触发；
2. 测量模块启动，完成指定的测量后将值写入片上 RAM（见34页）并引发中断；
3. CPU 被唤醒，从 RAM 取出测量值并处理，随后写 Command.ResRead 寄存器清空结果；
4. 等待下一个测量事件。

测量事件的触发有两种方式：

1. 单次测量：软件往 StartOP 位写 1，启动 MOP 类型的测量；
2. 定时测量：开启巡航模式，硬件会根据配置，周期性地产生测量事件。

测量是非常精密的过程，对环境要求较高，因此软件务必在测量模块工作前将芯片置为休眠模式，关闭处理器系统。

3.2 电源管理

龙芯 1D100BA 实现了完善的电源管理，通过有效的软件控制，可大大延长电池寿命。按电源状态分组，龙芯 1D100BA 的电压域包括：常开域（PMU、SLCD、RTC、RAM 等）、CPU 域（处理器子系统）、SNRTHS 域（时差/温度 AFE、TDC）、MBUS 域（串口 1）等如图3.1 所示。

除常开域外，其它模块均实现了低功耗关断模式。

MBUS 域的电源由 ChipCtrl.MbusOff 一位控制，写 1 关断，写 0 打开。关断后再次打开需要重新配置其寄存器。MBUS 域中用到 32K 时钟，解复位相对较慢。从上电到可以使用，软件需要等待 5 个 32K 时钟周期。在不使用期间（关断到上电可用前），模块的输出（UART1_TX）不确定。如果外部设备要求确定的状态，则软件应当在这段时间内将其配置为 GPIO。

SNRTHS 模块的电源主要由测量状态机自动控制。当有新的测量操作需要进行时，电源将自动打开；当测量结束后，电源自动关断。该模块的配置寄存器不会因为电源关断而丢失，因此软件无需反复配置。但应注意，如果其电源处于关断状态，模块内所有寄存器将无法访问。将 ChipCtrl.SonarOff 寄存器写 0 可强行将该模块打开，不受测量

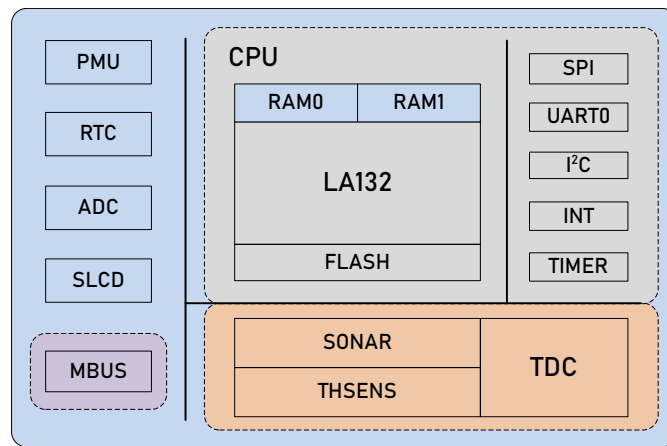


图 3.1: 电源架构

状态机的控制。SNRTHS 模块需要一段时间上电，软件强行打开后等 8 个时钟周期（8 条指令的延迟）才能访问。SonarOff 寄存器默认值为 0，为了降低功耗，软件在初始化完成后应将其写 1。

CPU 模块的关断由 Command.Sleep 控制，软件写 1 后进入关断模式。当 PMU 有中断要送到 CPU 时，CPU 模块将被唤醒。唤醒后 CPU 模块内所有内部寄存器都会被复位，处理器的执行与复位一致，Command.RstSrc 寄存器提供复位来源信息，软件可判断是一次外部复位，或者看门狗复位，还是一次待机唤醒。

板上集成的 SPI Flash 通常也有低功耗模式，需要在系统设计时加以考虑。较新的 SPI Flash 在不访问时可自动降低功耗，但有的 SPI Flash 还需要用发专门的命令进入或退出低功耗模式。龙芯 1D100BA 支持硬件自动发退出命令 (ChipCtrl.SpiLpEn)，但进入命令需要由软件发起。

3.3 巡航模式

龙芯 1D100BA 支持硬件自发进行的周期性测量，称为巡航模式。该模式下测量由硬件发起，测量结束后再唤醒软件进行结果处理，可有效地降低电流消耗。巡航模式下硬件可以自动地对调测量方向，不需要软件打开 SONAR 域电源来修改配置寄存器，详见 Timing.CruAutoDir。

巡航模式下测量事件的触发与 32.768K 时钟产生的 1/16 秒事件同步。根据巡航的时间间隔，当 Count 寄存器的低几位为 0，且即将加 1 时，触发测量事件。以每秒 4 次测量为例，比较时看 Count.RTC 的最低 6 位。若软件希望使用定时器唤醒，为了不干扰测量，最好将唤醒时间配置与测量事件错开。比如 Count 计数精度为 1/256 秒，测量事件在 8ms 内可保证完成，定时唤醒所使用的 Compare 值低 2 位为 2' b10 就可以保证与测量错开。

时差巡航和温度巡航的测量事件有可能同时产生，但时差巡航优先级高，只有测完时差，结束读出后，测量状态机才会接着处理温度测量。

若测量结果长时间未被读出，后续的巡航测量将无法开始。使用巡航模式时，软件应当被设计成能够及时处理测量结果。因此当出现连续两次测量未能执行时通常意味着系统进入了异常状态，软件可配置 Command.WatchEn，使芯片在碰到这种情况时进行看门狗复位。

3.4 寄存器定义

电源管理模块的基址为 0x100。

表 3.1: PMU 寄存器列表

名称	偏移	描述
Timing	0x00	测量时序控制
Command	0x04	测量命令、配置
Count	0x08	时间计数器
Compare	0x0c	唤醒时间配置
ChipCtrl	0x10	芯片全局配置
AutoCalc	0x14	自动计算配置
AdcCtrl	0x18	ADC 配置
AdcData	0x1c	ADC 数据
ExintEn	0x20	外部中断使能
ExintPol	0x24	外部中断极性
ExintEdge	0x28	外部中断边沿
ExintSrc	0x2c	外部中断状态
WdtCfg	0x30	看门狗配置
WdtFeed	0x34	看门狗重置端口
PowerCfg	0x38	电源配置
CommandW	0x3c	命令写端口
GPIOA_OE	0x40	GPIOA 输出使能
GPIOA_O	0x44	GPIOA 输出电平
GPIOA_I	0x48	GPIOA 输入电平
GPIOB_OE	0x50	GPIOB 输出使能
GPIOB_O	0x54	GPIOB 输出电平
GPIOB_I	0x58	GPIOB 输入电平
Pulse0	0x60	脉冲输出配置 0
Pulse1	0x64	脉冲输出配置 1
UserDat	0x68	用户数据
GPIOBit	0x80 ~ 0xbf	GPIO 位访问

3.4.1 时序控制 (Timing)

偏 移: 0x00

复位值: 32'h0

表 3.2: 时序控制

位域	名称	访问	描述
31	C8mFail	RO	8M 外部时钟失效 0: 正常 1: 失效
30	C8mSts	RO	8M 时钟选择状态 0: 选内部时钟 1: 选外部时钟
29	C32kFail	RO	32K 外部时钟失效 0: 正常 1: 失效
28	C32kSts	RO	32K 时钟选择状态 0: 选内部时钟 1: 选外部时钟
27	C8mSel	RW	CPU 运行时钟选择 0: 内部时钟 1: 外部 8MHz 晶体振荡器时钟 变化沿起作用。当外部时钟失效时, 将自动切换到内部时钟, 但此位保持不变。 当 CPU 唤醒时, 如果 Osc8mEn 和 C8mSel 均为 1, 则在等待 StartClkHS 时间后时钟会自动切换到外部
26	Osc8mEn	RW	高速晶体振荡器使能 测量期间自动打开振荡器, 此处仅对 CPU 运行的状态有效。 0: 关闭高速晶振 1: 开启高速晶振
25	C32kSel	RW	32K 时钟选择 0: 内部时钟 1: 外部时钟 变化沿起作用。当外部时钟失效时, 将自动切换到内部时钟, 但此位保持不变。
24	C32kSpeed	RW	内部 32K OSC 速度 0: 32K 1: 1K
23:20	C32kTrim	RW	内部 32K OSC Trimming 值 频率调节码, 用于控制内部 32K 时钟的频率。内部时钟频率无法做准, 仅作为永不停止的后备时钟使用。流量测量应用中软件应当切换到外部 32K 时钟, 否则定时的基准将无法保证。 0 ~ 15
19	CruDir	RO	方向对换状态 读数为测量方向有无自动对调。 0: 软硬件方向相同 1: 硬件使用的方向与软件在寄存器配置值相反 (PulseDef.FireSel/ PulseDef.StopSel 或 ThsensCtrl.PortDir)

位域	名称	访问	描述
18	CruAutoDir	RW	巡航模式下方向自动对调 0: 不自动对调 1: 自动对调, 软件可以从 Command.CruDir 读出本次结果的对调状态
17	CycleZero	RW	两次测量的间隔置零 0: 正常, 按 CycleTOF/Temp 配置 1: 置零, 两次测量连续进行
16	CruiseX	RW	快速巡航模式 0: 正常 1: 间隔为 CruiseTOF/Temp 的 1/4
15:14	CruiseTemp	RW	温度巡航的测量间隔 0: 2 秒 1: 8 秒 2: 32 秒 3: 128 秒
13:12	CruiseTOF	RW	时差巡航的测量间隔 0: 0.25 秒 1: 0.5 秒 2: 1 秒 3: 2 秒
11:10	CycleTemp	RW	往返两次温度测量的间隔 间隔为 50/60Hz 的倍数 0: 1 1: 1.5 2: 2 3: 2.5
9:8	CycleTOF	RW	往返两次超声时差测量的间隔 间隔为 50/60Hz 的倍数 0: 1 1: 1.5 2: 2 3: 2.5
7:6	CalResNum	RW	校准陶瓷晶振的 32K 时钟周期数 0: 2 周期 1: 4 周期 2: 8 周期 3: 16 周期
5	HZ60	RW	市电频率选择 0: 50Hz 1: 60Hz
4:2	StartClkHS	RW	高速晶振开启到开始测量的延迟。 0: 120us 1: 240us 2: 480us 3: 1.46ms 4: 2.44ms 其它: 5.14ms

位域	名称	访问	描述
1:0	ClkHSDiv	RW	测量参考时钟分频 0: 1 分频 1: 2 分频 2: 3 分频 3: 4 分频

3.4.2 测量命令、配置 (Command)

偏 移: 0x04

复位值: 32'h0

该寄存器有一些写 1 有效的只写位，这些位在偏移为 0x3c 处有另一个写端口 (CommandW)。往该端口写 1 等价于将 Command 读出，或上要写 1 的位再写回。

表 3.3: 测量命令、配置

位域	名称	访问	描述
31	ResRead	WO	结果读取标志 写 1 表示完成测量结果读取，硬件可进行新的测量。该操作只会复位结果指针，不改变结果结果存储
30:29	reserved	—	保留
28	FireComp	RW	互补发波使能 PT1、PT2 引脚复用为发波引脚，分别发出与 FIRE1、FIRE2 反相位的脉冲。如果同时进行温度测量，则 PT3、PT4 作为仅有的 2 个测温端口进行温度测量。 0: 关闭互补发波 1: 开启互补发波
27:26	RstSrc	RO	复位来源 00: 外部复位 01/10: 看门狗复位，每次看门狗复位均切换 11: 休眠唤醒
25	ExtIntEn	RW	外部唤醒使能 对应外部中断（见第七章）
24:16	IntSrc	RO	中断状态 0: e_Wake 1: e_ResValid 2: e_BatFail 3: e_C32kFail 4: e_C8mFail 5: e_Key 6: e_Ring 7: e_RTC 8: e_ExInt 往 CommandW 寄存器的对应位写 1 可清除中断状态

位域	名称	访问	描述
15:8	IntEn	RW	中断使能，每一位对应一个中断源 0: e_Wake 1: e_ResValid 2: e_BatFail 3: e_C32kFail 4: e_C8mFail 5: e_Key 6: e_Ring 7: e_RTC
7:6	CruiseMode	RW	巡航模式选择 0: 关闭巡航 1: 时差巡航 2: 温度巡航 3: 时差 + 温度巡航 巡航模式下时差和温度测量均为两次测量 MOP=2/3
5	WatchEn	RW	巡航模式看门狗使能 0: 关闭 1: 开启，巡航模式下连续两次时差或者连续两次温度测量未被服务则触发
4:2	MOP	RW	操作码 0: 一次超声时差测量 1: 一次温度测量 2: 往返两次超声时差测量，间隔为 CycleTOF 3: 往返两次温度测量，间隔为 CycleTemp 4: TDC 校准 5: 高频晶振校准 6: 换能器电容检测 读数为当前操作的操作码，巡航模式下操作码由硬件自动设置
1	StartOP	WO	启动测量操作 写 1 启动测量，操作码见 MOP
0	Sleep	WO	进入休眠状态 写 1 关闭处理器子系统

3.4.3 时间计数器 (Count)

偏 移: 0x08

复位值: 32' h0

表 3.4: 时间计数器

位域	名称	访问	描述
31:20	reserved	-	保留
19:0	RTC	RO	时间计数器 每 1/256 秒加 1

3.4.4 唤醒时间配置 (Compare)

偏 移: 0x0c

复位值: 32'h0

表 3.5: 唤醒时间配置

位域	名称	访问	描述
31:20	reserved	—	保留
19:0	WakeCmp	RW	唤醒时间配置 当该值与 Count 寄存器相等且 Wake 中断使能时产生唤醒事件

3.4.5 芯片全局配置 (ChipCtrl)

偏 移: 0x10

复位值: 从 SPI 启动为 32'h7dff, 否则为 32'h7fff

表 3.6: 芯片全局配置

位域	名称	访问	描述
31	PTTest	RW	PT 引脚输入测试模式 此位为 1, 且 PT* 复用为 GPIO 时进入 PT 引脚测试模式。该模式下 GPIO00 将输出 PT[1:4] 和 SenseT 引脚的异或值
30	TurboEn	RW	CPU 加力模式 1: 当 CPU 在 RAM 中执行指令时, 频率提高至 32MHz 0: 关闭 仅在内部时钟时有效
29	FastEn	RW	时钟提速使能 使用内部时钟的情况下, CPU 域上电复位后默认 4 分频工作。如果此位写 1, 则提升到 3 分频工作。写 1 后再写 0 不会立即将时钟降低为 4 分频, 而是要等下一次休眠唤醒才生效。 0: 4 分频 1: 3 分频
28	reserved	—	保留
27	SonarOff	RW	测量模块低功耗模式使能 0: 保持上电 1: 根据测量状态自动控制
26	MbusOff	RW	MBUS 模块电源关闭 0: 开启电源 1: 关闭电源
25	Pulse1Carr	RW	使用 Pulse1 为 UART0 的发送信号提供载波 0: 关闭, 使能 Pulse1.enable 后, 脉冲从引脚 PULSE1 输出 1: 打开, 使能 Pulse1.enable 后, 引脚 PULSE1 仍为 GPIO, 脉冲或在串口的输出数据上
24:23	BatdetSel	RW	掉电检测引脚选择 0: ADC_I0 (此时 adci0_ien 应为 1) 1: EMP_I 2: GPIO0(GPIO0 应为输入) 3: GPIO1(GPIO1 应为输入)

位域	名称	访问	描述
22	SonarOut	RW	测量控制输出 0: 下列引脚为 SPI 主功能 1: 引脚输出时差测量控制信号 SPI_CLK : SONAR_ACT SPI_MISO : SONAR_DIR SPI_MOSI : SONAR_RECV SPI_CSN0 : SONAR_SEND SPI_CSN1 : SONAR_FIRE
21	DebugOut	RW	测量调试信号输出 0: EMP_I 和 KEY_I 引脚为主功能 1: EMP_I 和 KEY_I 引脚输出 DBG0 和 DBG1 调试信号, 详见第26页 TdcCtrlL.Dbg[0:1]Sel
20	KeyOneInt	RW	按键中断模式 0: 按下和释放都发中断 1: 仅发按下中断, KEY_I 由高变低
19	SpiLpEn	RW	SPI Flash 退出低功耗模式使能 0: 关闭, CPU 上电时不自动读 SPI Flash 的 ID, 开启更迅速。 1: 打开, CPU 上电时通过读 ID 自动退出低功耗模式。软件负责在 CPU 掉电前将 SPI Flash 置为低功耗模式。
18	SpiFastUp	RW	SPI Flash 低功耗模式退出等待时间选择 0: 8us 1: 256us
17	EmpiPu	RW	EMPI 上拉使能 1: 上拉 20kohm
16	EmpiPd	RW	EMPI 下拉使能 1: 下拉 20kohm
15	Gpio0IUart0	RW	GPIO00/01 复用为 UART0 GPIO00: UART0_RX GPIO01: UART0_TX 写 1 使能
14	LcdAllGpio	RW	LCD_SEG00..19/LCD_COM0..3 模式选择 0: LCD 模式 1: GPIO 模式 此位置 1 后 LCD 控制器的时钟将被关断
13:10	LcdHiGpio	RW	LCD_SEG20..23 模式选择 0: LCD 模式 1: GPIO 模式 第 0 位对应 SEG20, 以此类推
9	SpiGpio	RW	SPI 接口复用为 GPIO
8	Pt12Gpio	RW	PT1、PT2 和 LoadT、SenseT 复用为 GPIO
7	Pt34Gpio	RW	PT3 和 PT4 复用为 GPIO
6	KeyiGpio	RW	KEY_I 复用为 GPIO
5	EmpiGpio	RW	EMP_I 复用为 GPIO

位域	名称	访问	描述
4	Uart1Gpio	RW	UART1 复用为 GPIO
3	Uart0Gpio	RW	UART0 复用为 GPIO
2	Adc1Gpio	RW	ADC_I1 复用为 GPIO
1	Adc0Ien	RW	ADC_I0 数字输入使能 0: 数字输入关闭, 只能用作电压测量 1: 数字输入打开, 可作掉电检测
0	I2cGpio	RW	I ² C 复用为 GPIO

3.4.6 自动计算配置 (AutoCalc)

偏 移: 0x14

复位值: 32' h0

自动计算过程描述详见4.3.3

表 3.7: 自动计算配置

位域	名称	访问	描述
31:29	reserved	—	保留
28:16	RamEntry	RW	RAM 程序入口地址 0 ~ 8K RAM 地址, 4 字节对齐 (低 2 位为 0)。
15:13	reserved	—	保留
12:8	HitNum	RW	要计算的结果数 应为 SonarCtrl.HitIn 值减 1
7:6	reserved	—	保留
5:4	FltRate	RW	滤波配置 控制 CPU 滤波指令的行为, 表示去掉几个最大值和几个最小值后累加。 0 ~ 3
3	RamBoot	RW	RAM 启动 无论是否有双向时差结果需要计算, 休眠唤醒后均从指定的 RAM 程序入口处执行。
2	EndGoRam	RW	自动计算结束后跳转 RAM
1	BegGoRam	RW	自动计算开始前跳转 RAM 不进行自动计算, 由 RAM 中指令处理。
0	CalcEn	RW	自动计算使能 0: 关闭自动计算 1: 若 CPU 休眠状态下被双向时差测量结束中断唤醒, 且结果个数符合预期, 则进行自动计算

3.4.7 ADC 配置 (AdcCtrl)

偏 移: 0x18

复位值: 32' h0

表 3.8: ADC 配置

位域	名称	访问	描述
31:8	reserved	–	保留
7	SnrChn	RW	时差测量通道模式 控制时差测量中进行 AD 转换所使用的通道。 0: 根据 AfeCtrl 控制自动选择 1: 使用 Sel 指定的通道
6:4	Sel	RW	通道选择 ADC 测量端口的选择 0: ADC_I0 1: ADC_I1 2: VRCAP 3: 1.0V 4: 无 5: 无 6: VREF 7: DAC/AMC
3	reserved	–	保留
2:1	Div	RW	时钟分频选择 0: 2 分频 1: 4 分频 2: 8 分频 3: 无效值 ADC 最高工作时钟为 4.5MHz，当使用内部时钟 3 分频作为系统时钟时，应当使用 4 分频模式。
0	Run	RW	启动单次测量 1: 启动 ADC 测量，结束后自动清零，如果开 ADC 中断，会收到中断

3.4.8 ADC 数据 (AdcData)

偏 移: 0x1c

复位值: 32' h0

表 3.9: ADC 数据

位域	名称	访问	描述
31:12	reserved	–	保留
11:0	Data	RO	ADC 测量数据

3.4.9 外部中断使能 (ExintEn)

偏 移: 0x20

复位值: 32' h0

表 3.10: 外部中断使能

位域	名称	访问	描述
31:0	En	RW	外部中断使能 0 ~ 15 位: GPIOA0 ~ GPIOA15 16 ~ 31 位: GPIOB0 ~ GPIOB15

3.4.10 外部中断极性 (ExintPol)

偏 移: 0x24

复位值: 32'h0

表 3.11: 外部中断极性

位域	名称	访问	描述
31:0	Pol	RW	外部中断极性 对应关系同上 0: 高电平/上升沿有效 1: 低电平/下降沿有效

3.4.11 外部中断边沿 (ExintEdge)

偏 移: 0x28

复位值: 32'h0

表 3.12: 外部中断边沿

位域	名称	访问	描述
31:0	Edge	RW	外部中断边沿模式选择 0: 电平模式 1: 边沿模式, 对电平宽度无要求

3.4.12 外部中断状态 (ExintSrc)

偏 移: 0x2c

复位值: 32'h0

表 3.13: 外部中断状态

位域	名称	访问	描述
31:0	Src	RO	外部中断状态 对应关系同上
31:0	Clr	WO	边沿模式中断清除 写 1 清除边沿模式记录下的中断状态, 电平模式中断应当关闭中断使能 或通过其它办法让外设不再发中断

3.4.13 看门狗配置寄存器 (WdtCfg)

偏 移: 0x30

复位值： 32' hffffb0004

看门狗配置必须满足低 16 位奇数个” 1”，且高 16 位为低 16 位取反的要求，否则会立即复位，如默认复位值为看门狗配置 4 秒等待延迟。看门狗无法关闭，只有调试模式会暂停计数。

表 3.14: 看门狗配置寄存器

位域	名称	访问	描述
31:16	wdtcfg_hi	RW	看门狗配置高位 应当为 wdtcfg_lo 取反
15:0	wdtcfg_lo	RW	看门狗配置低位 看门狗复位等待时间，以 1 秒为单位，最高位为奇偶校验位。

3.4.14 看门狗重置寄存器 (WdtFeed)

偏 移： 0x34

复位值： 32' h0

表 3.15: 看门狗重置寄存器

位域	名称	访问	描述
31:0	WdtFeed	WO	看门狗重置 写入 0xa55a55aa 将看门狗计数器重置为初始值，写入其它值无效

3.4.15 电源配置 (PowerCfg)

偏 移： 0x38

复位值： 32' h0

表 3.16: 电源配置

位域	名称	访问	描述
31:29	vr0_tctrim	RW	VR0 电压调节参数，硬件自动配置，不建议更改
28:24	vr0_abstrim	RW	VR0 电压调节参数，硬件自动配置，不建议更改
23:1	reserved	—	保留
0	vr0_stb_snr	RW	VR0 时差测量期间模式设置 0: 时差测量期间 VR0 为 active 模式，供电 20mA 1: 时差测量期间 VR0 为 standby 模式，供电 2mA

3.4.16 命令写端口 (CommandW)

偏 移： 0x3c

复位值： 32' h00000000

表 3.17: 命令写端口

位域	名称	访问	描述
31	ResRead	WO	结果读取标志 写 1 表示完成测量结果读取，硬件可进行新的测量。该操作只会复位结果指针，不改变结果结果存储
23:16	IntClr	WO	中断状态清除 0: e_Wake 1: e_ResValid 2: e_BatFail 3: e_C32kFail 4: e_C8mFail 5: e_Key 6: e_Ring 7: e_RTC 写 1 清除中断状态，外部中断的状态清除见表3.13
1	StartOP	WO	启动测量操作 写 1 启动测量，操作码见 MOP
0	Sleep	WO	进入休眠状态 SleepEn 为 1 时，写 1 则处理器系统关闭
0	SleepEn	RO	可进入休眠状态 当读出值为 1 表示可休眠

3.4.17 GPIOA 输出使能 (GPIOA_OE)

偏 移: 0x40

复位值: 32' h0

对应 0 ~ 31 号 GPIO 引脚，参考数据手册引脚定义。

表 3.18: GPIOA 输出使能

位域	名称	访问	描述
31:0	Oe[i]	RW	GPIOA 引脚的输出使能 0: 输入 1: 输出 管脚对应关系参见数据手册

3.4.18 GPIOA 输出电平 (GPIOA_O)

偏 移: 0x44

复位值: 32' h0

表 3.19: GPIOA 输出电平

位域	名称	访问	描述
31:0	Out[i]	RW	GPIOA 引脚的输出电平 0: 低电平 1: 高电平

3.4.19 GPIOA 输入电平 (GPIOA_I)

偏 移: 0x48

复位值: 32' h0

表 3.20: GPIOA 输入电平

位域	名称	访问	描述
31:0	In[i]	RO	GPIOA 引脚的输入 0: 低电平 1: 高电平

3.4.20 GPIOB 输出使能 (GPIOB_OE)

偏 移: 0x50

复位值: 32' h0

对应 32 ~ 58 号 GPIO 引脚, 参考数据手册引脚定义。

表 3.21: GPIOB 输出使能

位域	名称	访问	描述
31:0	Oe[i]	RW	GPIOB 引脚的输出使能 0: 输入 1: 输出 管脚对应关系参见数据手册

3.4.21 GPIOB 输出电平 (GPIOB_O)

偏 移: 0x54

复位值: 32' h0

表 3.22: GPIOB 输出电平

位域	名称	访问	描述
31:0	Out[i]	RW	GPIOB 引脚的输出电平 0: 低电平 1: 高电平

3.4.22 GPIOB 输入电平 (GPIOB_I)

偏 移: 0x58

复位值: 32' h0

表 3.23: GPIOB 输入电平

位域	名称	访问	描述
31:0	In[i]	RO	GPIOB 引脚的输入 0: 低电平 1: 高电平

3.4.23 脉冲输出配置 (Pulse0/1)

偏 移: 0x60/0x64

复位值: 32'h0

表 3.24: 脉冲输出配置

位域	名称	访问	描述
31:18	reserved	—	保留
17	Enable	RW	脉冲输出使能 0: 关闭, PULSE* 引脚为 GPIO 模式 1: 打开, 根据配置输出占空比 50% 的脉冲信号 在改变时钟选择和 (或) 分频系数前应将使能关闭, 完成配置后再打开
16	ClkSel	RW	时钟源选择 0: 32K 1: clkxs
15:0	PulseDiv	RW	脉冲分频系数 1~ 65535: 1~ 65535 分频

3.4.24 用户数据 (UserDat)

偏 移: 0x68

复位值: 无

表 3.25: 用户数据

位域	名称	访问	描述
31:0	user_dat	RW	用户数据 复位不会清除

3.4.25 GPIO 位访问端口 (GPIOBit)

偏 移: 0x80 ~ 0xbf

复位值: 无

此处有 64 个字节的空間, 前 59 个字节分别对应 59 个 GPIO。需用字节访问, 每个字节内的定义如下

表 3.26: GPIO 位访问端口

位域	名称	访问	描述
1	Oe	WO	GPIO 方向 1 为输出
0	In	RO	GPIO 输入
0	Out	WO	GPIO 输出

第四章 时差测量

4.1 时差测量流程

时差测量包括以下步骤：

1. 测量事件触发，ClkHS 开启；
2. 等待 StartClkHS 指定的时间后，时差测量相关模块开启；
3. STOP 通道充电至参考电位；
4. 脉冲发生器产生超声脉冲，从所选 FIRE 通道输出；
5. 拉零阶段将发送端 STOP 强下拉；
6. 等待阶段结束后进入收波阶段，所选 STOP 通道被选通，经过窗口使能后送到 TDC 进行测量。

图4.1 给出了整个测量过程以及相关配置参数的说明。

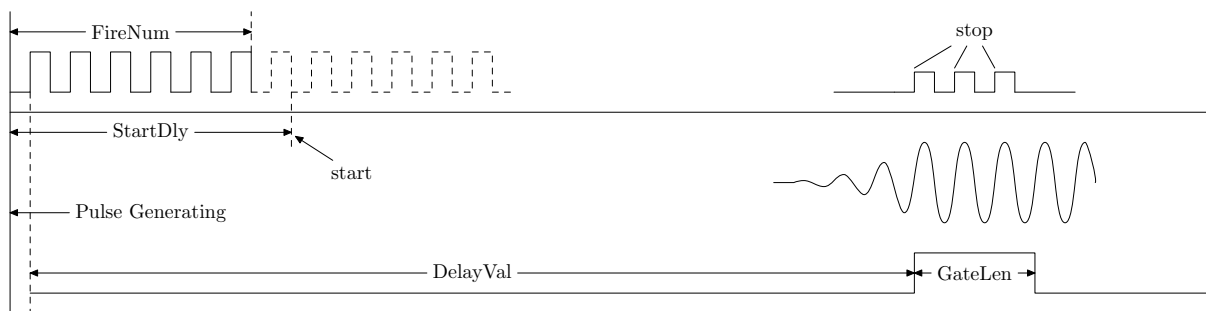


图 4.1: 时差测量过程

为了方便第一个波的定位，龙芯 1D100BA 实现了首波模式（图4.2）。其测量流程与普通模式只有窗口的设置存在差异：

1. 窗口 1 用于定位首波，此时比较器偏移与普通模式相同；
2. 首波发现后，窗口 1 结束，比较器偏移归零；
3. 窗口 2、3 以检测到首波下降沿后，参考时钟一个周期内开始计数，TDC 测量在窗口使能区间的回波信号。

4.2 时差测量寄存器

时差测量模块的基地址为 0x200 。

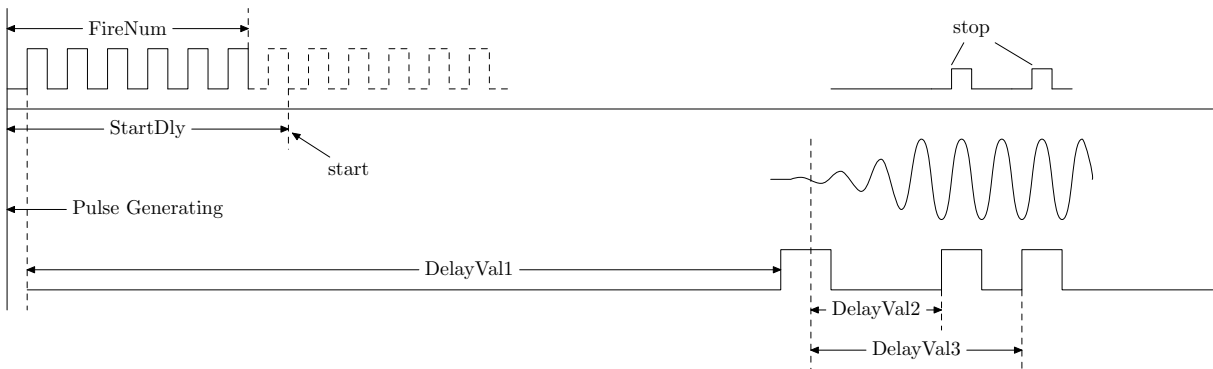


图 4.2: 首波模式时差测量

表 4.1: SONAR 寄存器列表

名称	偏移	描述
PulseDef	0x00	脉冲定义
SonarCtrl	0x04	时差测量控制
TdcCtrl1	0x08	TDC 控制 1
TdcCtrl2	0x0c	TDC 控制 2
CalibCtrl	0x10	校准控制
FWaveCtrl	0x14	首波控制
AfeCtrl	0x18	模拟前端控制
DLineCtrl	0x1c	窗口延迟链配置
GateDef1	0x20	窗口配置 1
GateDef2	0x24	窗口配置 2
GateDef3	0x28	窗口配置 3
ResPtr	0x2c	结果指针

4.2.1 发送脉冲定义 (PulseDef)

偏 移: 0x00

复位值: 32' h0

表 4.2: 发送脉冲定义

位域	名称	访问	描述
31	FireInv	RW	Fire 时钟取反 0: 不取反 1: 取反
30:16	FirePhase	RW	发送脉冲相位向量 16 ~ 30 位对应第 1 ~ 15 个脉冲, 第 15 波之后的相位无法设置, 统一为先高后低, 每位的相位定义为 0: 先高后低 1: 先低后高
15	StopSel	RW	Stop 接收端选择 0: Stop1 1: Stop2
14	FireSel	RW	Fire 发送端选择 0: Fire1 1: Fire2
13:7	FireDiv	RW	发送脉冲分频系数 1 ~ 127: 2 ~ 254 分频
6:0	FireNum	RW	发送脉冲个数 0: 关闭 1 ~ 127: 1 ~ 127 个脉冲

4.2.2 时差测量控制 (SonarCtrl)

偏 移: 0x04

复位值: 32' h0

表 4.3: 时差测量控制

位域	名称	访问	描述
31:30	TimeOutSel	RW	溢出时间选择 0: 64us 1: 256us 2: 1024us 3: 4096us
29:28	TRefChrg	RW	将 Stop 输入充电到参考电压的时间 0: 90us 1: 120us 2: 150us 3: 300us 此时间值应小于 StartClkHS 所配置值

位域	名称	访问	描述
27:26	HitIgn	RW	忽略 Stop 个数 当超声频率高于 TDC 的转换速度时，可以通过此参数屏蔽部分 Stop，使测量速率与 TDC 速率匹配。窗口开启后第一个 Stop 不会被忽略。 0: 连续 1 ~ 3: 1 ~ 3 个 Stop
25:20	HitIn	RW	预期 Stop 脉冲数 不包含结果将存储在状态区的脉冲 0: 关闭 1 ~ 32: 1 ~ 32 个脉冲
19	WatchFire	RW	发波观测 Fire 发波期间测量回波，测量控制电路跳过发波结束判断，直接进入测量阶段。需要与其它时序控制 (WaitEnd/TailEnd) 配合，且避免做自动前校准。 0: 关闭发波观测 1: 开启发波观测
18	CalNegEdge	RW	负沿校准 相比于正沿校准，负沿校准可以节省少量功耗。 0: 正沿校准 1: 负沿校准
17	AutoCalPst	RW	TDC 自动后校准 完成预期个数的 Stop 测量后进行校准。 0: 关闭 TDC 后校准 1: 开启 TDC 后校准
16	AutoCalPre	RW	TDC 自动前校准 开始发波后的第 10 个 ClkHS 时钟周期开始校准。 0: 关闭 TDC 前校准 1: 开启 TDC 前校准
15	NegStop	RW	Stop 通道反相 TDC 默认测量上升沿，如需测 Stop 下降沿则配置负沿触发。 0: 正沿触发 1: 负沿触发
14:8	TailEnd	RW	拉零阶段结束时间 以发波周期为单位，零点为第一个波。 0 ~ 127
7:0	WaitEnd	RW	等待阶段结束时间 以发波周期为单位，零点为第一个波。 0 ~ 255

4.2.3 TDC 控制 1 (TdcCtrl1)

偏 移: 0x08

复位值: 32'h0

表 4.4: TDC 控制 1

位域	名称	访问	描述
31:28	reserved	-	保留

位域	名称	访问	描述
27:26	DbglSel	RW	DBG1 引脚调试输出选择 0: 内部的 Stop 1: 比较器输出 2: TDC 窗口使能 3: LDSENSET 施密特输入
25:24	Dbg0Sel	RW	DBG0 引脚调试输出选择 0: 内部的 Start 1: 内部的 Stop 2: 比较器输出 3: TDC 窗口使能
23:20	reserved	—	保留
19	RingFix	RW	环误差修正使能 0: 关闭 1: 开启
18:15	Dlytune	RW	延迟量调节 0 ~ 7 (0 最大, 7 最小)
14	CorrSub	RW	修正方法 0: 加法 1: 减法
13	CalhitSel	RW	校准时钟沿 0: 前校准用第一个, 后校准用第二个 1: 前校准用第二个, 后校准用第一个
12	CaloneHit	RW	校准期间只测单个值 0: 关闭 1: 开启
11	Calmr1	RW	校准期间使用测量方式 1 0: 关闭 1: 开启
10	Caltype	RW	校准双采样方式 0、1
9	Calsmp	RW	校准双采样模式 0: 关闭 1: 开启
8	Calres	RW	校准双精度模式 0: 关闭 1: 开启
7	CpdSame	RW	校准与测量使用相同预延迟值 0: 关闭 1: 开启
6	CpdEn	RW	校准期间预延迟使能 0: 关闭 1: 开启
5	PdAuto	RW	预延迟自累加 0: 关闭 1: 开启

位域	名称	访问	描述
4	PdEn	RW	预延迟使能 0: 关闭 1: 开启
3	reserved	–	保留
2	Dstype	RW	双采样方式 0: 时钟负沿 1: 自定时
1	Dsmp	RW	双采样模式 0: 关闭 1: 开启
0	Dres	RW	双精度模式 0: 关闭 1: 开启

4.2.4 TDC 控制 2 (TdcCtrl2)

偏 移: 0x0c

复位值: 32' h0

表 4.5: TDC 控制 2

位域	名称	访问	描述
31	PdEnd	RO	预延迟量到极值的标志 0: 当前预延迟量加上增量后已经超出 1 个 ClkHS 时钟周期 1: 延迟量未到极值
30:26	reserved	–	保留
25:16	PdAuto	RO	自动维护的预延迟量 从 0 开始, 每次双向时差测量结束后加一个增量, 到达极值后自动归零。 0 ~ 1023
15:14	reserved	–	保留
13:12	PreDlyInc	RW	预延迟量增量 自动维护时递增的步长 0: +1 1: +2 ...
11:10	reserved	–	保留
9:0	PreDlyVal	RW	预延迟量 启动预延迟功能时生效 0 ~ 1023

4.2.5 零点校准控制 (CalibCtrl)

偏 移: 0x10

复位值: 32' h0

表 4.6: 零点校准控制

位域	名称	访问	描述
31:26	reserved	–	保留
25:16	ZeroLevel	RO	校准结果零点值 0 ~ 1023
15	reserved	–	保留
14:12	CalDelay	RW	校准环路延迟 校准期间设置一个新码到结果稳定可供判断的延迟。
11	CalBypass	RW	关闭自动校准 0: 开启自动校准 1: 关闭自动校准
10	CalMethod	RW	校准算法 0: 线性搜索 1: 二分查找
9:0	CalZero	RW	软件设置零点值 关闭自动校准时，直接使用此值作为零点；开启自动校准且使用线性搜索算法时，使用此值作为初始参数。 0 ~ 1023

4.2.6 首波控制 (FWaveCtrl)

偏 移: 0x14

复位值: 32' h0

表 4.7: 首波控制

位域	名称	访问	描述
31:30	reserved	–	保留
29:20	FHL2	RW	首波阈值 2 在反向回波等待期间使用的比较器偏移，用于定位首波。 0 ~ 1023
19:17	reserved	–	保留
16	WavePEn	RW	波峰测量使能 测量首波之后回波的峰值电压，配合第二窗口和 AfeCtrl 使用。 0: 关闭波峰测量 1: 开启波峰测量
15:14	reserved	–	保留
13: 4	FHL	RW	首波阈值 在回波等待期间使用的比较器偏移，用于定位首波。如果使能 FHL2，FHL 仅用于正向回波的首波定位。 0 ~ 1023
3	reserved	–	保留
2	FHL2En	RW	首波阈值 2 使能 0: 关闭 FHL2 1: 开启 FHL2
1	reserved	–	保留

位域	名称	访问	描述
0	AutoFW	RW	首波模式使能 0: 关闭首波模式 1: 开启首波模式

4.2.7 模拟前端控制 (AfeCtrl)

偏 移: 0x18

复位值: 32'h0

表 4.8: 模拟前端控制

位域	名称	访问	描述
31	VTestSel	RW	ADC 测量通道选择 0: VREF 1: DAC 或 AMC
30	VTestDual	RW	双向时差测量都启动 ADC 0: 仅在后一次测量启动 1: 两次都启动
29	VTestEn	RW	ADC 测量使能 0: 关闭 ADC 测量 1: 时差测量结束后启动 ADC 测量 ADC 测量结果写入状态区 0xledc
28	reserved	—	保留
27	ManMode	RW	AFE 内部模块人工控制 正常测量过程, 仅在必要的阶段开启相关模块。设置此位后将屏蔽模块自动关闭功能, 由相关控制位直接控制。LNA 例外, 其仅在首波定位后开启。 0: 自动控制 1: 人工控制
26:25	LdRef	RW	LNA 输入端驱动控制 00: 高阻 01: 发波前拉至 VREF, 随后高阻 10: 发波后拉至 VREF 11: 一直拉至 VREF
24	RxEn	RW	RX 接收使能 0: 关闭 RX 回波信号接收 1: 开启 RX 回波信号接收, 时差测量时必须置 1, 零点校准期间会自动关闭
23:22	TxDrive	RW	发波驱动能力 00: 无效值 01: 阻抗 85Ω 10: 阻抗 40Ω 11: 阻抗 27Ω

位域	名称	访问	描述
21:20	RxPullDown	RW	发波端的 RX 强下拉 00: 不强拉 01: 尾波期间强拉 10: 无效值 11: 整个测量期间 (排除发波) 强拉
19	StopTerm0	RW	接收端拉零 0: 接收端 TX 输出 VREF 1: 接收端 TX 输出 0 电压
18	EOPIEn	RW	EOPI 使能 0: EOPI 引脚悬空 1: 方向选择后的 RX 回波信号与 EOPI 引脚连通 (外部放大器输入端)
17	VCMPEn	RW	VCMP 使能 0: VCMP 引脚悬空 1: 内部比较器的参考电压与 VCMP 引脚连通
16:14	VTestOut	RW	模拟电压控制 1 选择内部模拟电压输出至 ADC xx0: 关闭输出 011: VCMP 比较器参考电压 (DAC) 101: VAMC 波峰电压 其他: 无效值
13	VRefOut	RW	模拟电压控制 2 选择内部模拟电压输出至 ADC 0: 关闭输出 1: 输出参考电压 VREF
12	EOPOEn	RW	EOPO 使能 0: EOPO 引脚悬空 1: 片内比较器输入端与 EOPO 引脚连通 (外部放大器输出端)
11:10	CompSel	RW	比较器控制 00: 断开 LNA, 使用片外放大器 01: 旁路 LNA, RX 回波信号直接输入比较器 10: 使用 LNA, 比较器直接用于首波检测, 检出首波后, LNA 自动从旁路状态切换到使用状态。 11: 无效值
9	CompEn	RW	比较器使能 0: 关闭比较器, 前提为 RX 是全摆幅数字输入 1: 开启比较器, 一般时差测量期间必须置 1
8	LnaEnc	RW	LNA 电容反馈 0: 关闭电容反馈 1: 开启电容反馈
7	LnaEn	RW	LNA 使能 0: 关闭 LNA 1: 开启 LNA
6:5	VRefDrv	RW	VREF 驱动能力 00: 无效 01: 阻抗 95Ω 10: 阻抗 68Ω 11: 阻抗 59Ω

位域	名称	访问	描述
4	VRefEn	RW	VREF 使能 0: 关闭 VREF 1: 开启 VREF, 时差测量期间必须置 1
3	VRefSel	RW	VREF 选择 0: 0.70V 1: 0.88V
2	DacUgEn	RW	DAC 单位增益放大器使能 0: 关闭增益放大器 1: 开启增益放大器, 时差测量期间必须置 1
1	DacRange	RW	DAC 范围 0: 0 ~ 0.8V 1: 0.54 ~ 1.08V
0	DacEn	RW	DAC 使能 0: 关闭 DAC 1: 开启 DAC, 时差测量期间必须置 1

4.2.8 窗口延迟链配置 (DLineCtrl)

偏 移: 0x1c

复位值: 32' h0

表 4.9: 窗口延迟链配置

位域	名称	访问	描述
31:15	reserved	—	保留
14:8	DLineLen	RO	校准结果 1 个 ClkHS 时钟周期的链长。
7:2	reserved	—	保留
1	DlCalByp	RW	延迟链校准值旁路 0: 使用校准值 1: 旁路校准值, 此时窗口配置的 5 位小数部分与 ClkHS 周期无关, 以 16ns 为单位的延迟量, 且此值随环境变化。
0	DlCalEn	RW	延迟链校准使能 0: 关闭校准 1: 时差测量前启动延迟链校准

4.2.9 窗口配置 (GateDef)

偏 移: 0x20, 0x24, 0x28

复位值: 32'h0

表 4.10: 窗口配置

位域	名称	访问	描述
31	GateKeep	RW	窗口保持打开 0: 打开 GateLen 指示的周期数后关闭 1: 保持打开, 不再考虑下一个窗口配置
30	OneShot	RW	发生单次 hit 后关闭当前窗口 延迟不超过 500ns
29:28	reserved	—	保留
27:20	GateLen	RW	窗口打开长度, 以 ClkHS 为单位 0 ~ 255: 1 ~ 256 周期
19:0	DelayVal	RW	20 位延迟量, 以 ClkHS 为单位, 15 位整数, 5 位小数 {Int15.Frac5} 窗口延迟量说明: 1) 延迟值为 Int15+2+Frac5/32 个 ClkHS 周期; 2) 非首波模式下 3 个窗口的时间零点都为 Fire1 或 Fire2 的上升沿; 3) 首波模式下 GateDef2/3 的时间零点在首波下降沿的 0.5 ~ 1 个 ClkHS 周期内; 4) 整数部分不能为 0, 后一个窗口打开时间与前一个窗口结束之间需保留 2 个周期。

4.2.10 结果指针 (ResPtr)

偏 移: 0x2c

复位值: 无

表 4.11: 结果指针

位域	名称	访问	描述
31:24	st1	RO	反向测量的状态指针 反向测量是指 fire_sel/port_dir 为 1 的测量 指针以 Word 为单位
23:16	st0	RO	正向测量的状态指针 正向测量是指 fire_sel/port_dir 为 0 的测量 指针以 Word 为单位
15: 8	res1	RO	反向测量的数据指针
7: 0	res0	RO	正向测量的数据指针

4.3 测量结果处理

4.3.1 结果存储

测量结果按一定的格式存储在数据 RAM 中，其解读与 TDC 配置和测量目标有关。

表 4.12: 测量结果存储

结果区域	基址	大小 (Words)	说明
RES0	0x1f80	32	正向测量数据区
RES1	0x1f00	32	反向测量数据区
ST0	0x1ee0	8	正向测量状态区
ST1	0x1ec0	8	反向测量状态区

测量结果存储分为 4 个区域，如表4.12 所示。分别由四个写指针维护写入偏移。在上电复位以及清结果 (ResRead) 后，写指针初始化为所在区域的最低地址。随着测量结果的写入，写指针不断递增。由于每个测量结果是 4 字节，写指针也以 4 字节为单位递增。表4.11 中存储的是以 Word 为单位的值，代表已写入的测量结果个数。

在测量结果中，Start 值、TDC 校准值归类为状态，存储于状态区；其它结果存储于数据区。状态区的结果个数与测量配置相关，最少 1 个，最多 5 个。状态区的最后一个 Word(0x1efc) 存储测量结束时的写指针状态字。软件可直接从 RAM 中读出，而不需要给 SNRTHS 模块供电再从 ResPtr 寄存器读出。值得注意，状态区的状态字最高位为测量超时位，其为 1 时表示预设的时间内没有测到期望的结果个数。状态区 0x1edc 所存储 ADC 转换的结果，其中低 16 位和高 16 位分别对应正向和反向测量。

状态区测量结果排列如表4.13 所示。表格中排列假定所有测量选项均打开，当部分选项关闭时，引入的空位将由后续测量项补上。比如在时差测量中仅开启 TDC 后校准，将有 3 个状态，后校准的两个结果存储于偏移 1 和 2 处。

表 4.13: 状态区测量结果排列

次序	时差测量	温度测量	TDC 校准	晶振校准	电容检测
0	start	pstcal1	cal1	pstcal1	pstcal1
1	precal1	pstcal2	cal2	pstcal2	pstcal2
2	precal2				
3	pstcal1				
4	pstcal2				

4.3.2 结果计算

TDC 的测量以 ClkHS 时钟为参考。当内部的 start 开始后，粗值计数器从 0 开始计数。之后的每个测量的结果除了 TDC 的输出外还有距离 start 的周期数。以测 1 个 stop (HitIn=1) 为例，测量量与测量结果的对应关系如图4.3 所示。 FC_{start} 和 FC_{stop1} 分

别为 TDC 测量 start 和 stop 信号到内部参考时钟的时差， CC_{stop1} 为以参考时钟周期计数的粗值计数器的值。从 start 信号到 stop 信号的时间差为：

$$\Delta T = T_{ClkHS} * \left(CC_{stop1} + \frac{FC_{start} - FC_{stop1}}{CAL_2 - CAL_1} \right) \quad (4.1)$$

其中， FC_* 位于 TDC 输出测量结果的低 17 位， CC_* 位于高 15 位， CAL_1 和 CAL_2 为一次校准所产生的前后两个值。校准所产生的结果不记录 CC 值，只有低位 FC 有效。

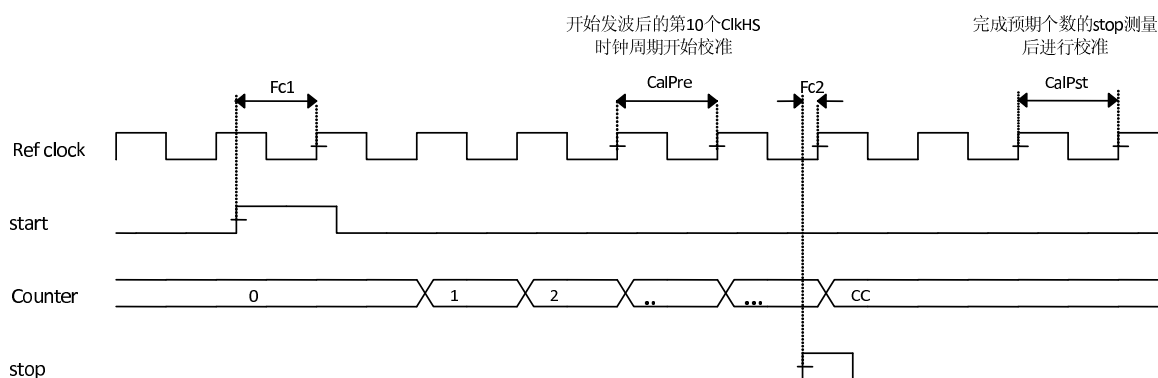


图 4.3: 测量过程示例

4.3.3 时差测量自动计算

流量表中最为频繁出现的事件是双向时差测量，当采样比较多时，其计算会消耗掉可观的 CPU 周期。为加速这个过程，龙芯 1D100BA 设计了自动计算功能。当 CPU 被双向时差测量结果中断唤醒，且测量结果有效未超时，则启动自动计算。其操作是将测量结果存储区的数据对逐个进行时间转换、相减（反向减去正向），并随后滤波累加。以 22 个测量点，去 3 大 3 小为例，完成自动计算后得到 16 个测量数据的和，存放在地址 0x1eb8 处。计算结果表示为 32 位定点小数，高 15 位为整数部分，低 17 位为小数部分。整数 1 表示一个整周期 (ClkHS) 时长。软件后处理时再把结果除以滤波后数据个数，得到最终的时差值。

自动计算时不做校准，软件需要将 TDC 校准值（24 位整数 C ）提前写入地址 0x1ebc。TDC 细值计数的单位时长是整周期粗值计数单位时长的 $2^{-36}C$ 。

自动计算除了将结果写入地址 0x1eb8 外，不产生或改变其它任何状态。软件可以在取走结果后将其写成特殊值，从而判断是否有新结果写入。同理，软件还要写 Command[W].ResRead，通知硬件这次测量的结果已经读走。

4.3.4 校准测量

由于 TDC 的测量结果与温度、电压等外部环境相关，测量结果必须进行校准。校准以高频时钟 ClkHS 为参考，得到 TDC 单位分辨率所对应的时间。TDC 校准产生两个结果，分别为一个和两个时钟周期的测量值，软件需将它们相减，得到一个时钟周期所对应的计数值。

4.3.5 高频晶振校准

与石英晶体相比，使用陶瓷振荡器有助于减少起振时间，降低电流消耗。但陶瓷振荡器的频率稳定性较差，需要基于 32.768K 的石英晶体进行校准，得到更精确的高频时钟周期值。高频晶振通过对 32.768K 时钟计数产生的两个事件进行测量实现校准。两个事件的间隔由 CalResNum 寄存器设置。

4.4 脉冲发生器

脉冲发生器用于向 FIRE 端口送出可控制的脉冲，并给 TDC 一个开始信号。在工作时，高频时钟 ClkHS 先经过分频产生目标时钟，然后经相位处理送至指定的 FIRE 端口，通过 FireNum 参数控制输出脉冲个数。TDC 开始信号与 FIRE 端送出的第一个脉冲上升沿同步。

4.5 窗口与波峰测量

精确测量飞行时间的关键是选择合适的被测波，龙芯 1D100BA 支持首波模式，即先用相对大的比较偏移，定位首个跨过该阈值的回波（首波），然后将比较偏移归零，并以首波为窗口时间参考点采样后续的波。龙芯 1D100BA 有 3 个时间窗口，非首波模式下，3 个窗口的时间零点均为发出第一个波的上升沿；首波模式的窗口 1 被用来定位首波，检出首波下降沿的一个参考时钟周期内复位窗口时间计数器，窗口 2、3 分别重新计数。窗口可以配置为单次触发或固定窗口长度两种模式，单次触发模式为窗口内出现一次被测事件即关闭窗口。

为了更好的选取首波阈值（首波比较偏移电压），龙芯 1D100BA 配备了波峰测量功能。波峰测量模式重新定义 3 个窗口的功能：1) 定位首波，2) 定位波峰测量结束点，3) 定位被测波。窗口 2 仅开启时间有意义，作为波峰测量采样结束时刻。波峰测量使用定制的 AMC 模块，从收波阶段开始跟随接收信号的最大电压值，直至窗口 2 的打开时间点，随后保持峰值电压直到完成 ADC 采样，如图4.4所示。注意只能测首波之后的波峰，如果想测靠前的回波，需要调低首波阈值，使首波位置前移。

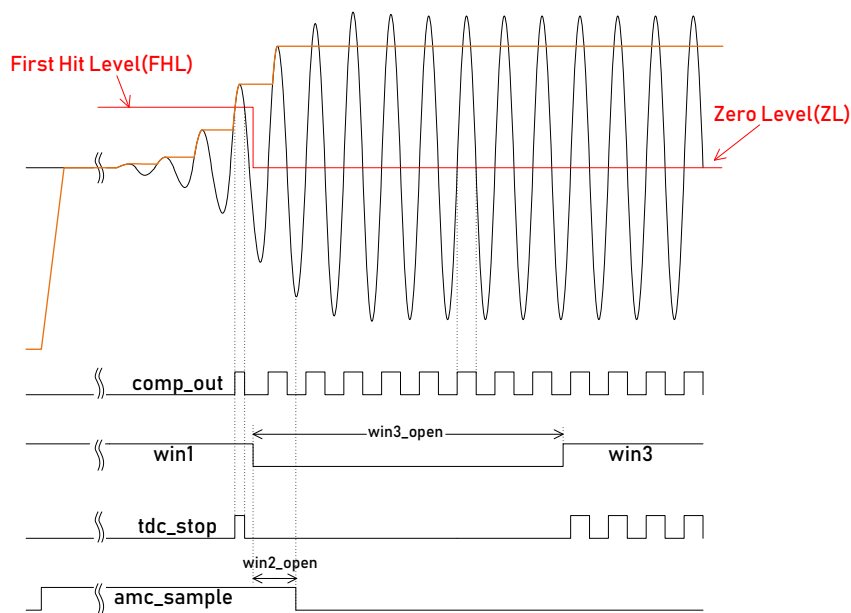


图 4.4: 波峰测量过程

4.6 测量功能拓展

龙芯 1D100BA 集成低噪放大器 (LNA)，其带宽 >3MHz，增益 20dB，增益带宽基 5.1MHz，满足超声波水表、热量表等典型应用场景需求。此外，龙芯 1D100BA 支持测量控制信号、回波信号输出，搭配外部高压发波驱动电路、收波放大器，满足外夹式水表与气体测量等应用需求。回波路径外接放大器如图4.5所示。超声波飞行时间测量的控制信号（参考 PMU 模块 ChipCtrl.SonarOut）如图4.6所示。

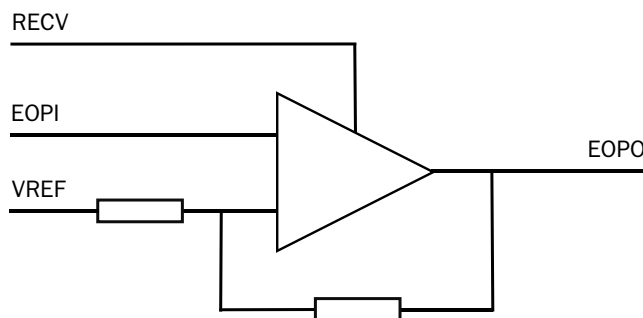


图 4.5: 外置放大器示意图

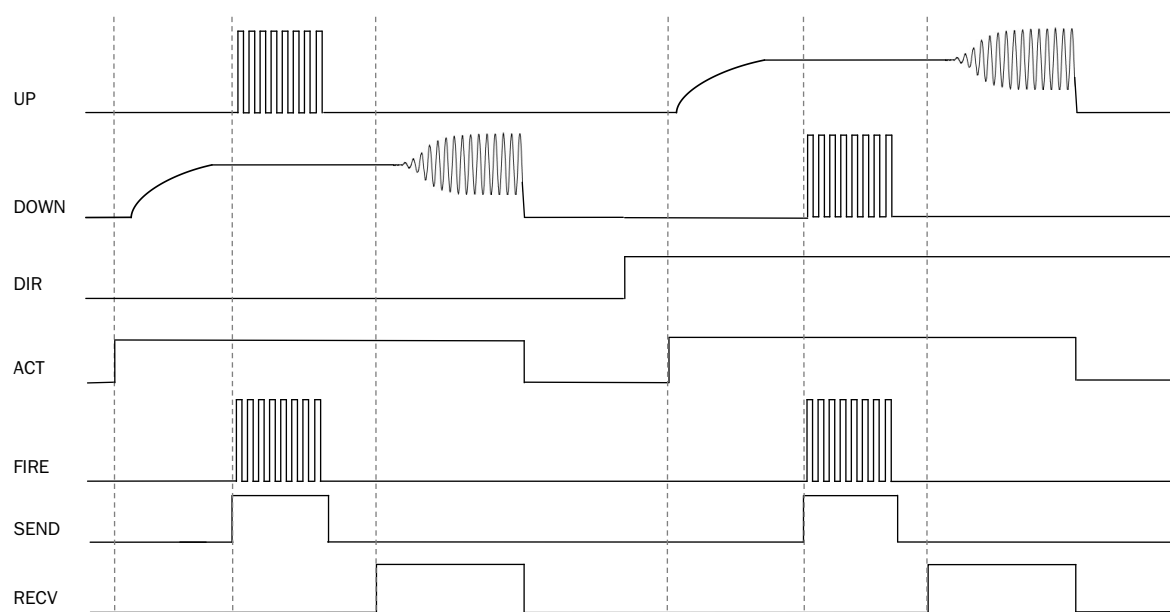


图 4.6: 测量控制信号

4.7 TDC 配置与输出

龙芯 1D100BA 集成了一个环形 TDC，输出宽度 17 位，代表所测量的事件到内部参考时钟上升沿所经过的时间（信号走过的内部精密延迟单元的个数）。该 TDC 实现了双精度和四精度模式，若测量与校准的配置相同，则软件不需要特殊处理。直接把单位延迟（TDC 1 个 LSB 所代表的时间）乘以计数值即可。否则，必须根据配置对校准结果进行转换，得到测量中实际的单位延迟。TDC 输出 17 位，除校准外，其高位拼接 15 位 ClkHS 周期计数一同写入测量结果存储区。

TDC 实现了预延迟功能，在测量事件发生后附加额外的一段确定的延迟，随后才测量到最近的参考时钟的时间。这段延迟不能超过一个 ClkHS 周期，否则有可能使延迟后的时间量超出 TDC 测量范围。在测量 start 事件时硬件会评估现有的预延迟是否即将超出一个 ClkHS 周期（SonarCtrl2.PdEnd 为 0）。通过调整预延迟可以引入一定的扰动，优化测量的效果。硬件中包含自动的延迟量维护逻辑，每个来回两次时差测量后会自动加 TdcCtrl2.PreDlyInc 所代表的增量值。

第五章 温度测量

5.1 温度测量流程

温度测量通过电容充放电的形式将温度量转换为时间量。每个测量点 (PT*) 产生两个结果，分别为放电开始时间和放电结束时间。软件需将每一对结果相减，得到电容放电时间长度，进而计算温敏电阻与参考电阻的阻值比例关系，求解温度值。如果结果个数少于测量点个数的 2 倍，则表明外部电路存在故障。测量流程包括若干个充放电周期 (图5.2)，每个周期由电容充电、放电两部分组成，如图5.1 所示。

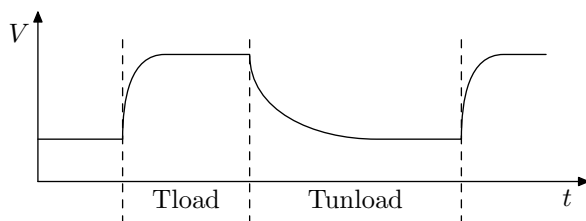


图 5.1: 温度测量充放电周期

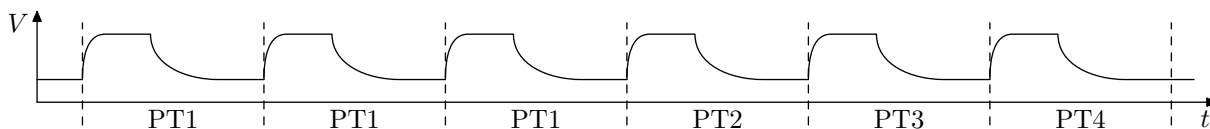


图 5.2: 一次温度测量流程

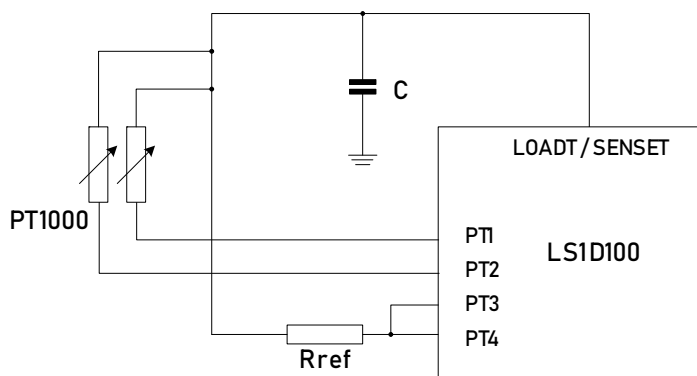


图 5.3: 温度测量电路连接

5.2 温度测量寄存器

温度测量模块的基地址为 0x280。

5.2.1 温度测量控制 (ThsensCtrl)

偏移: 0x00

复位值: 32'h0

表 5.1: 温度测量控制

位域	名称	访问	描述
0	FakeNum	RW	在温度测量前的热身假测量 0: 2 次 1: 7 次
1	PortNum	RW	测量的端口数 0: 2 个 (仅包含 PT1 和 PT2; PT1 和 PT2 复用为 FireComp 时, 仅包含 PT3 和 PT4) 1: 4 个
2	PortDir	RW	温度测量端口的测量顺序 0: PT1->PT2->PT3->PT4 1: PT4->PT3->PT2->PT1
3	reserved	-	保留
5:4	Tload	RW	充电时间长度 0: 90us 1: 120us 2: 150us 3: 500us
7:6	Tunload	RW	放电时间长度 0: 90us 1: 120us 2: 150us 3: 500us
8	FastMode	RW	快速模式 0: 放电时间由 Tunload 决定 1: 放电完成即开始下一个周期, Tunload 定超时时间

第六章 段式 LCD 控制器

6.1 概述

段式 LCD 控制器支持段式液晶屏，输出电压为 3 段式，提供更好的对比度。该控制器最多支持 96 段，包含 4 个公共输出 COM0-COM3 和 24 个段输出 SEG0-SEG23。每对 COM 和 SEG 输出控制 1 个段。

段式 LCD 控制器的基地址为 0x420。

6.2 寄存器定义

表 6.1: 段式 LCD 控制器寄存器列表

名称	偏移	描述
FB0	0x0	帧缓冲 0 寄存器
FB1	0x4	帧缓冲 1 寄存器
FB2	0x8	帧缓冲 2 寄存器
RFR	0xc	刷新率寄存器

6.2.1 帧缓冲 0 寄存器 (FB0)

偏 移: 0x0

复位值: 32'h0

表 6.2: 帧缓冲 0 寄存器

位域	名称	访问	描述
31:0	fb0	RW	帧缓冲 0 表示帧缓冲第 31 至 0 位是否显示, 1 为显示, 0 为不显示

6.2.2 帧缓冲 1 寄存器 (FB1)

偏 移: 0x4

复位值: 32'h0

表 6.3: 帧缓冲 1 寄存器

位域	名称	访问	描述
31:0	fb1	RW	帧缓冲 1 表示帧缓冲第 63 至 32 位是否显示, 1 为显示, 0 为不显示

6.2.3 帧缓冲 2 寄存器 (FB2)

偏 移: 0x8

复位值: 32' h0

表 6.4: 帧缓冲 2 寄存器

位域	名称	访问	描述
31:0	fb2	RW	帧缓冲 2 表示帧缓冲第 95 至 64 位是否显示, 1 为显示, 0 为不显示

6.2.4 刷新率寄存器 (RFR)

偏 移: 0xc

复位值: 32' h20

表 6.5: 刷新率寄存器

位域	名称	访问	描述
31:8	reserved	—	保留
7:0	refresh_rate	RW	刷新率 作为分频值的一部分用于得到合适的刷新率, 见使用说明

6.3 使用说明

6.3.1 帧缓冲排布

帧缓冲与 COM 和 SEG 的关系如下表所示:

表 6.6: 帧缓冲排布表

SEG	COM0	COM1	COM2	COM3	SEG	COM0	COM1	COM2	COM3
0	0	1	2	3	12	48	49	50	51
1	4	5	6	7	13	52	53	54	55
2	8	9	10	11	14	56	57	58	59
3	12	13	14	15	15	60	61	62	63
4	16	17	18	19	16	64	65	66	67
5	20	21	22	23	17	68	69	70	71
6	24	25	26	27	18	72	73	74	75
7	28	29	30	31	19	76	77	78	79
8	32	33	34	35	20	80	81	82	83
9	36	37	38	39	21	84	85	86	87
10	40	41	42	43	22	88	89	90	91
11	44	45	46	47	23	92	93	94	95

在使用时, 还应根据液晶屏的段和显示位置的表格再进行一次映射。

6.3.2 刷新率设置

图6.1给出了段式 LCD 的工作原理，COM 和 SEG 在段的两端给出不同电平，电平之差幅度大时有显示，幅度小时无显示。

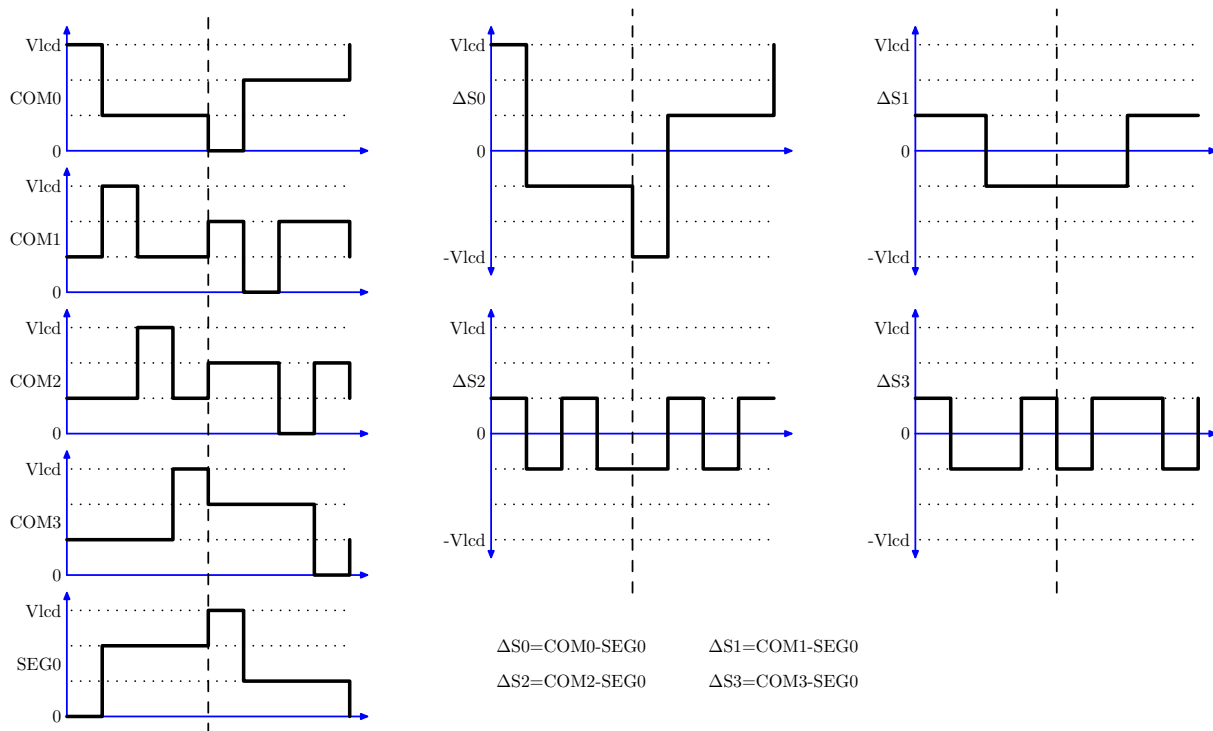


图 6.1: 段式 LCD 工作原理图

图中零点至虚线，即半个周期的长度定义为 1 帧，每秒帧数为 $\frac{1024}{refresh_rate}$ ，因此默认的帧数为 32。提高帧数有助于减少闪烁，提高液晶寿命，但也会增加功耗。

此页留空

表 7.1: 中断对应关系

中断号	中断名	说明
IP7	ExInt	来自 GPIO 的中断，参见表 3.10~ 3.13 寄存器定义。
IP6	INTC	详见本章寄存器定义。
IP5	RTC Ring Key	RTC 定时中断 UART1(MBUS) 中断 KEY_I 按键中断，可配置为仅按下或者是按下和释放都中断
IP4	C8mFail C32kFail BatFail	外部 8M 时钟异常 外部 32K 时钟异常 电池掉电
IP3	ResValid	测量结果有效，每次测量结束都会产生这个事件。
IP2	Wake	PMU 唤醒中断，参见 PMU.Count（表3.4）和 PMU.Compare 寄存器。
IP1	SoftInt	软件中断，写 1 置中断状态。具体用法参见处理器核手册。
IP0	SoftInt	软件中断，写 1 置中断状态。具体用法参见处理器核手册。

7.2 中断处理

当某控制器需要使用中断方式时，需打开整个中断连接路径上的所有中断使能，包括处理器 CSR.ECFG 寄存器的中断使能位，PMU 或者 INTC 中的中断使能位，对应控制器的中断使能位（如果存在）。CPU 的中断处理入口由 CSR.EENTRY 配置，需要由软件区分中断源并进行相应的中断处理。中断源为脉冲型的由中断控制寄存器保存状态，清中断时只需清除中断控制寄存器对应位即可。中断源为电平型的则根据中断控制器的配置有所不同，若配置中断控制器配成电平模式，则只需服务中断源对应的模块，清除中断原因；若中断控制器配成边沿模式，则除了服务中断源对应模块外，还应清除中断控制寄存器的对应位。ExInt 的中断类型可配置，其它 PMU 汇总的中断均为边沿模式。

中断处理的基本流程如下：

- 将寄存器值保存到 SRAM 中。
- 读取处理器 CSR.ESAT 寄存器，获取 IP7-IP0 的中断状态，结合 CSR.ECFG 寄存器中的中断使能位，获知处理器中断源。
- 根据表7.1继续查询中断控制器和模块相关寄存器，获知中断发生原因。
- 依据不同中断源进行相关处理，包括事件处理和清中断。
- 处理完成后恢复寄存器值。
- 返回正常执行流。

7.3 寄存器定义

中断相关寄存器存放于 conf 模块，其基地址为 0x400。

表 7.2: Conf 寄存器列表

名称	偏移	描述
INT_EN	0x00	中断使能寄存器
INT_EDGE	0x01	中断边沿寄存器
INT_POL	0x02	中断极性寄存器
INT_CLR	0x03	中断清除寄存器
INT_SET	0x04	中断置位寄存器
INT_OUT	0x05	中断输出寄存器
SRPROT	0x06	运行状态及保护寄存器

7.3.1 中断使能寄存器 (INT_EN)

偏 移: 0x00

复位值: 8' h0

表 7.3: 中断使能寄存器

位域	名称	访问	描述
7	reserved	—	保留
6	adc_int_en	RW	ADC 中断使能位 置 1 使能
5	spi_int_en	RW	SPI 中断使能位 置 1 使能
4	flash_int_en	RW	Flash 中断使能位 置 1 使能
3	uart0_int_en	RW	UART0 中断使能位 置 1 使能
2	uart1_int_en	RW	UART1 中断使能位 置 1 使能
1	i2c_int_en	RW	I ² C 中断使能位 置 1 使能
0	timer_int_en	RW	定时器中断使能位 置 1 使能

7.3.2 中断边沿寄存器 (INT_EDGE)

偏 移: 0x01

复位值: 8' hff

除外部中断外，其他中断的边沿位由硬件设计决定，软件不应对其进行修改

表 7.4: 中断边沿寄存器

位域	名称	访问	描述
7:0	int_edge	RW	中断边沿位，对应于中断使能寄存器的各位 置 1 表示边沿触发，置 0 表示电平触发

7.3.3 中断极性寄存器 (INT_POL)

偏 移: 0x02

复位值: 8' hff

除外部中断外, 其他中断的极性位由硬件设计决定, 软件不应对其进行修改

表 7.5: 中断极性寄存器

位域	名称	访问	描述
7:0	int_pol	RW	中断极性位, 对应于中断使能寄存器的各位 置 1 表示高电平/上升沿触发

7.3.4 中断清除寄存器 (INT_CLR)

偏 移: 0x03

复位值: 8' h0

写 1 后该位自动清零, 无需手动清零

表 7.6: 中断清除寄存器

位域	名称	访问	描述
7:0	int_clr	WO	中断清除位, 对应于中断使能寄存器的各位 置 1 清除内部中断状态

7.3.5 中断置位寄存器 (INT_SET)

偏 移: 0x04

复位值: 8' h0

写 1 后该位自动清零, 通常不使用, 仅作为测试中断

表 7.7: 中断置位寄存器

位域	名称	访问	描述
7:0	int_set	WO	中断置位位, 对应于中断使能寄存器的各位 置 1 置位边沿触发模式的内部中断状态

7.3.6 中断输出寄存器 (INT_OUT)

偏 移: 0x05

复位值: 8' h0

表 7.8: 中断输出寄存器

位域	名称	访问	描述
7:0	int_out	RO	中断输出位, 对应于中断使能寄存器的各位 值为 1 表示中断触发, 仅在对应中断使能后触发

7.3.7 运行状态及保护寄存器 (SRPROT)

偏 移: 0x6

复位值: 8'h0

表 7.9: 运行状态及保护寄存器

位域	名称	访问	描述
7	addr_check_en	RW	地址检查使能 1 表示进行地址检查, 当软件发出未定义的地址时触发 NMI 中断 往此寄存器连续写入 0x00, 0x5a, 0xa5 打开此位写使能
6	reserved	–	保留
5	jtag_lock	RO	JTAG 锁定 1 表示 JTAG 接口被禁用
4	otp_lock	RO	OTP 锁定 1 表示 OTP 区域被禁写
3	new_pkg	RO	新封装模式 表示上电配置非封装兼容模式
2	jtag_func	RO	JTAG 复用 1 表示 JTAG 接口可复用为 GPIO
1	install_mode	RO	安装模式 1 表示当前为安装模式
0	boot_spi	RO	SPI 启动 1 表示当前从 SPI 启动

此页留空

第八章 Flash

8.1 概述

龙芯 1D100BA 提供了大小为 128KB 的片内 Flash，可用于程序和数据的存储，Flash 包含 1024 个页，每个页 128 个字节。

Flash 的地址空间包含两个部分：存储空间和寄存器空间。存储空间的基地址为 0x1f000000，大小为 128KB。当系统从 Flash 启动时，0x1c000000 地址也可访问存储空间。寄存器空间的基地址为 0x500，包含 6 个 32 位寄存器。

Flash 还包含一个特殊的 OTP 页用于保存特殊配置值，其访问地址在 128KB 以上，即从 0x1f020000 开始的一个页。该页的最高 4 字节为控制两个配置位，分别表示 OTP 锁定和 JTAG 锁定。

8.2 存储空间

存储空间与 Flash 的 128KB 存储空间直接对应，可直接读取。

当需要修改 Flash 的内容时，需要进行擦写操作，一次完整的修改流程如下：

1. 清 page_latch
2. 写有效值至目标地址
3. 配置擦除时间
4. 擦除目标页
5. 配置编程时间
6. 编程目标页

上述流程中提及的 page_latch 是 Flash 内部的一个暂存空间，大小与一个页相同。每次清空后，只能对其中每个字节写入一次，否则可能导致数据错误。对存储空间的写操作将直接修改 page_latch，故写操作的有效范围为一个页。一次擦写前的存储空间写入值应都在一个页中，否则将导致写入错误。对存储空间的写操作只支持 32 位写。

当执行擦除命令时，会将目标页全部擦除；当执行编程命令时，会将 page_latch 中的值编程至 Flash 中。

8.3 寄存器空间

表 8.1: Flash 控制器寄存器列表

名称	偏移	描述
CMD	0x00	命令寄存器
STS	0x14	状态寄存器
PET	0x18	擦写时间寄存器
BOUND	0x34	BOUND 寄存器
OTPCFG	0x38	OTP 配置寄存器

8.3.1 命令寄存器 (CMD)

偏 移: 0x00

复位值: 32' h0

表 8.2: 命令寄存器

位域	名称	访问	描述
31:28	command	RW	命令 命令完成后自动清零。 4' b0011 : 清中断 4' b0100 : 清 page_latch 4' b1001 : 更新区域保护 4' b1010 : 擦除目标页 4' b1100 : 进入休眠模式 4' b1110 : 编程目标页
27:18	reserved	—	保留
17:0	pageaddr	RW	目标页地址 擦除或编程的目标页地址, 范围为 128KB

8.3.2 状态寄存器 (STS)

偏 移: 0x14

复位值: 32' h0

表 8.3: 状态寄存器

位域	名称	访问	描述
31:4	reserved	—	保留
3	no_permission	RO	无权限 中断状态为无权限, 表示上一次命令无操作权限
2	pe_end	RO	擦写结束 中断状态为擦写结束, 表示擦写命令完成
1:0	reserved	—	保留

8.3.3 擦写时间寄存器 (PET)

偏 移: 0x18

复位值： 32' h10

表 8.4: 擦写时间寄存器

位域	名称	访问	描述
31:18	reserved	—	保留
17:16	int_en	RW	中断使能 由高位至低位对应于 no_permission、pe_end
15:6	reserved	—	保留
5:3	etime	RW	擦除时间 0:1.5ms 1:2.0ms 2:2.5ms 3:3.0ms 4:3.5ms 5:4.0ms 6:4.5ms 7:5.0ms 以 8MHz 时钟计算，默认的擦除时间为 2.5ms
2:0	ptime	RW	编程时间 0:1.5ms 1:2.0ms 2:2.5ms 3:3.0ms 4:3.5ms 5:4.0ms 6:4.5ms 7:5.0ms 以 8MHz 时钟计算，默认的编程时间为 1.5ms

8.3.4 BOUND 寄存器 (BOUND)

偏 移： 0x34

复位值： 32' h1000

表 8.5: BOUND 寄存器

位域	名称	访问	描述
31:16	reserved	—	保留
15:0	bound	RO	区域保护 bound 寄存器复位为 ISP 段地址 0x1000，更新 bound 命令有效时变为命令中包含的地址，flash 复位后取出 special row1 中保存的值

8.3.5 OTP 配置寄存器 (OTPCFG)

偏 移： 0x38

复位值： 32' h0

表 8.6: OTP 配置寄存器

位域	名称	访问	描述
31:16	otp_lock_cfg	RO	OTP 锁定配置 只读，用于查看当前 otp_lock 配置位的数据
15:0	jtag_lock_cfg	RO	JTAG 锁定配置 只读，用于查看当前 jtag_lock 配置位的数据

8.4 使用说明

8.4.1 代码保护

为了防止误擦写带来的风险，Flash 实现了一种代码保护的机制。

Flash 的存储空间分为三段： 0 ~ 4K 为 ISP 固化代码段， 4K ~ bound 为代码

段，bound ~ 128K 为数据段。ISP 段只能由 JTAG 和 SPI 启动模式改写，代码段只能由 JTAG、SPI 启动和 ISP 程序改写，数据段没有限制。

不符合上述规则的擦写将无法完成并产生 no_permission 中断。bound 地址可通过更新区域保护命令进行配置，该地址值将存入特殊页并自动读出，bound 无法通过软件读取。

8.4.2 中断

Flash 控制器中包含两种中断：pe_end 和 no_permission，中断的状态可通过状态寄存器查询获得。pe_end 表示擦写结束，no_permission 表示该次擦写无权限。

在中断处理结束后，需要进行清中断。命令寄存器中的清除中断命令将同时清除两种中断。

8.4.3 编程指南

—— 读取 Flash 中内容 ——

```
temp = *(volatile unsigned int*)(0x1f000000);
```

—— 擦写 Flash 中偏移为 0x100 的页 ——

```
PE_TIME = 0x30019;           // 配置擦写时间为 3/2ms
CMD      = 0xa0000100;       // 擦除命令
wait();           // 实现 wait 指令，等待中断唤醒
CMD      = 0x40000000;       // 清除 page_latch
for(i=0x1f000100; i<0x1f000180; i=i+4) // 128 字节
    *(volatile unsigned int*)(i) = 0x12345678; // 写入 page_latch
CMD      = 0xe0000100;       // 编程至 0x100 偏移地址
wait();
```

8.4.4 OTP 功能

OTP 页的最高两字节为 0x5aa5 时表示 OTP 锁定，Flash 复位后生效，锁定后 OTP 页只能读出，不可擦除或编程，也无法解除 OTP 锁定。OTP 页的次高两字节为 0xa55a 时表示 JTAG 锁定，Flash 复位后生效，锁定后芯片的 JTAG 功能只能读出 IDCODE，并且只能从片内 Flash 启动。OTP 页的其它区域可用于存储用户自定义的产品序列号信息。

第九章 电压测量

9.1 概述

龙芯 1D100BA 内置 12 位 (ENOB=9.5) 的模数转换模块 (ADC), 可用于电池电压检测和外部信号测量。该 ADC 的参考电压为 VDD, 输出的测量值为被测信号相对于参考电压的比值。假设被测电压为 V_{in} , 输出值 $dout[11:0] = V_{in}/VDD * 4096$ 。应注意被测电压应当在 0 到 VDD 之间。

ADC 有八个测量通道, 连接方式如下表所示。

表 9.1: ADC 测量通道

通道	连接	说明
0	ADC0_I	芯片引脚
1	ADC1_I	芯片引脚
2	VRCAP	芯片引脚, 数字部分电源电压
3	VR0_BG	数字供电 LDO 的 1.0V 基准电压
6	VREF	模拟前端参考电压
7	DAC/AMC	模拟前端内部信号

芯片的数字部分电源电压的典型值 (常温 25℃) 在出厂前会以 ACSII 码格式写入 0x1f0201a0 所指向的数据字中, 单位为 mV (例如 1985mV)。通过对 VRCAP 的测量, 可反向推导出 VDD 电压值。如果 VDD 与电池直接相连, 则电池电压可据此得到。

当 ADC_I0 用于送入模拟量给 ADC 测量时, 建议将 ADC_I0 管脚的输入使能关闭 (见 ChipCtrl.Adc0Ien 控制位), 避免不必要的漏电。

9.1.1 模拟前端电压测量

模拟前端内部电压只在其工作期间有效, 因此这部分测量控制由 SONAR 模块发起, 工作在外部 8M 时钟的二分频上。测量结果写入 0x1edc 和 0x1ede 指向的两个半字, 分别对应 FIRE1 发波和 FIRE2 发波的时差测量。使能模拟前端电压测量后, 时差测量过程将延长, 相应地增加部分功耗。如果不是必需, 不建议每次时差测量都打开 AfeCtrl.VTestEn 位。

当 AdcCtrl.SnrChn 为 1 时, 所测量的通道将不限于模拟前端内部, 比如可被用于测量外部电压。这个功能主要用于硅后验证, 正常情况下不需要使用。

9.2 寄存器定义

参见第17页表3.8和表3.9。

9.3 使用说明

AD 控制器可使用中断或轮询方式进行工作，AD 控制器的中断为脉冲式，其使能、触发等状态由 Confreg 中中断相关寄存器控制。ADC 单次测量需要约 18 个分频时钟的时间，对于 4 分频来说是 72 时钟周期。如果软件可接受这个延迟，则也可使用轮询方式，通过查询 AdcCtrl.Run 是否清零来判断测量完成。

下面的程序片段给出通过轮询方式进行多次测量的寄存器配置方法。

配置寄存器

```
ADC_CTRL = 0x11;           // 测量通道 1，时钟选择主时钟二分频
while(ADC_CTRL & 0x1);      // 查询 Run 位，直到被清零
res = ADC_DATA;             // 获取测量结果
```

第十章 I²C 控制器

10.1 概述

I²C(Inter-integrated Circuit) 总线接口连接微控制器和串行 I²C 总线。它提供多主机功能，控制所有 I²C 总线特定的时序、协议、仲裁和定时。支持标准和快速两种模式。龙芯 1D100BA 集成 1 路 I²C 控制器，其寄存器基地址为 0x480。

10.2 主要特性

- 多主机功能：该模块既可做主设备也可做从设备
- I²C 主设备功能
 - 产生时钟
 - 产生起始和停止信号
- I²C 从设备功能
 - 可编程的 I²C 地址检测
 - 停止位检测
- 产生和检测 7 位地址和广播呼叫
- 支持不同的通讯速度
 - 标准模式（高达 100 kHz）
 - 快速模式（高达 400 kHz）
- 状态标志：
 - 发送器/接收器模式标志
 - 字节发送结束标志
 - I²C 总线忙标志
- 错误标志
 - 主模式时的仲裁丢失
 - 地址/数据传输后的应答（ACK）错误
 - 检测到错位的起始或停止条件
 - 禁止拉长时钟功能时的上溢或下溢
- 1 个中断输出
 - 用于地址/数据通讯成功和错误
- 可选的拉长时钟功能
- 可配的 1~ 7 周期的数字滤波器

10.3 功能描述

10.3.1 模式选择

控制器可以工作在从发送、从接收、主发送、主接收等 4 种模式之一。其中无论主从，发送均表示将数据写到 SDA 线，接收均表示从 SDA 线获取数据。

控制器默认处于从模式，在主动发起“开始”命令或“恢复”命令时，自动切换到主模式；当仲裁丢失或总线出现“停止”条件时，自动切换回从模式。

控制器的时钟与 APB 总线时钟相同，当使用外部 8M 晶体时，为 8MHz；当使用内部时钟且使能时钟提速（PMU.FastEn=1）时，约为 10.6MHz（内部 32M 时钟进行 3 分频）。

在从模式下，需要配置时钟频率（CR2_FREQ），用于产生合适的建立和保持时间。而在主模式下，数据线的建立和保持时间总是为一半的时钟线低电平时间。

10.3.2 从模式

从模式下，当检测到“开始”条件时，从 SDA 线接收地址，接收完成后与从地址寄存器（OAR）或广播地址（当 CR2_ENGC 有效时）进行比对。

若地址不命中，控制器无视本次总线传输（不会进行应答）并等待下一个“开始”条件。

若地址命中，若允许应答（CR1_ACK 有效）则会产生 I²C 总线应答；产生应答后，硬件置起地址状态位（SR1_ADDR），若事件中断使能（CR2_ITEV TEN）有效，还会产生中断。

发送状态位（SR2_TRA）表示从设备处于发送还是接收状态。

10.3.2.1 从发送模式

从设备拉低 SCL 线，直到地址状态位清除且数据寄存器填入需要发送的第一个数据。填入第一个数据之前，数据寄存器处于发送空状态（SR1_TXE）。移位寄存器从数据寄存器取出第一个字节，开始移位并发送，取出后数据寄存器再次进入发送空状态。

当第一个发送的数据完成了应答后，若数据寄存器仍然处于空状态，则触发字节传输结束状态（SR1_BTF）并等待。在字节传输结束状态被清除前，控制器进入等待状态并拉低 SCL 线。

若完成应答后，数据寄存器不处于空状态，则取出下一个字节并发送。即在前一个数据完成传输前，若查询到 SR1_TXE 状态，即可写入数据寄存器以免进入 SR1_BTF 状态导致总线空闲。

若发送的数据接收到的应答为无应答（NACK），表示主设备不再接收数据，此时应答失败状态（SR1_AF）会置起，控制器回到空闲状态，等待下一个“开始”条件。

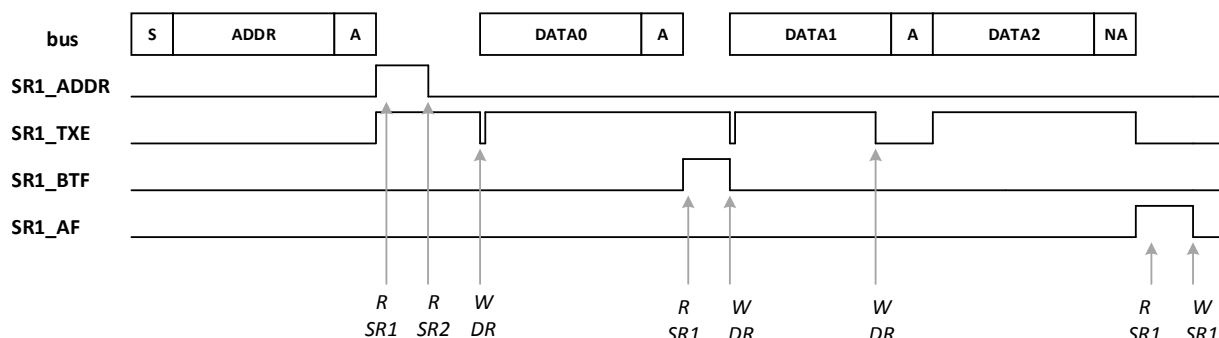


图 10.1: 从发送模式示意图

10.3.2.2 从接收模式

清除地址状态位后，从设备开始从 SDA 线接受数据放入移位寄存器。

收满一个字节后，若接收数据寄存器为空，则在应答周期完成后放入接收数据寄存器中，置起接收非空状态（SR1_RXNE），并继续进行下一个字节的接收。若应答配置（CR1_ACK）为无应答，则会结束本次传输，进入空闲状态等待下一个“开始”条件。

若接收数据寄存器不为空，则数据保留在移位寄存器中，并置起字节传输结束状态（SR1_BTF）并等待。在字节传输结束状态被清除前，控制器进入等待状态并拉低 SCL 线。

当主设备发出“结束”条件时，控制器进入空闲状态，并置起停止状态位（SR1_STOPF）。

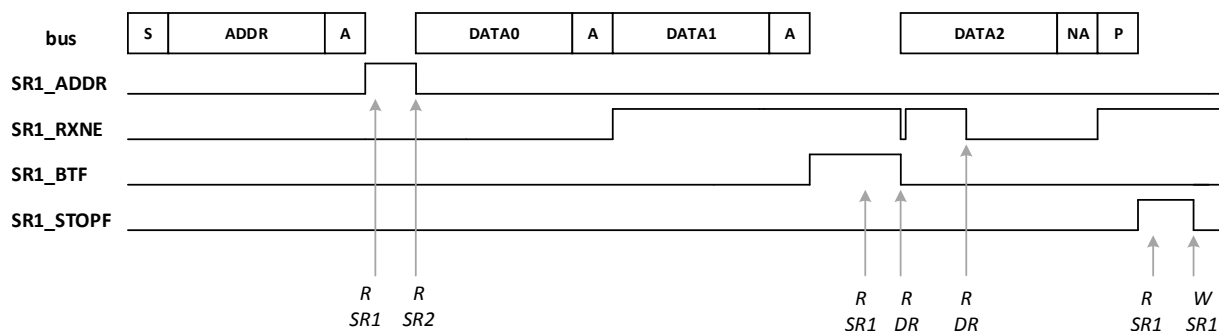


图 10.2: 从接收模式示意图

从接收模式存在一种可能：从设备还没有及时处理数据和停止条件，后续的传输就已经开始。这是由于单独的 SR1_RXNE 不会拉低 SCL，主设备可执行停止和新的开始。

需要注意，在图10.3.2.2中若最后一个字节未读取，则内部接收数据寄存器保持非空（即使 SR1_RXNE 已因写入 DR 或不在接收模式而被清除），在下一次接收模式时（无论主从），会直接产生 SR1_BTTF 状态。也就是说，内部维护的接收非空状态（非 SR1_RXNE）只能由读 DR 而清除。

若 CR1_STOPF 也没有处理，则后续传输中会看到 SR1_STOPF 和 SR1_ADDR 同时置起。此时，若前一次访问为接收，则接收数据寄存器有前一次传输的数据。

为避免接收数据混乱，建议软件有足够快的处理速度，并及时读出接收的数据；处理时，先处理完 SR1_RXNE 再处理 SR1_STOPF；

10.3.3 主模式

主模式下，控制器初始化数据传输并提供时钟信号（SCL）。

当内部状态机接收到“开始”命令（CR1_START）或“恢复”命令（CR1_RECOVER）时，即进入主模式，从而支持在总线开始阶段即开始主设备仲裁检测。

10.3.3.1 主模式时钟生成

由于主模式需要产生时钟，故需要一些额外的配置：

- 配置时钟控制寄存器（CCR）
- 配置上升时间寄存器（TRISE）

其中，CCR 配置不同的时钟占空比和分频系数。TRISE 用来确定 SCL 上升时间的上限，在主模式发送 SCL 的上升沿后，等待 TRISE 时间再检查 SCL 输入的状态；若此时 SCL 输入为低，则表示从设备因为无法处理而拉低了 SCL，此时主模式的时钟发送暂停，直到 SCL 恢复为高。

等待 TRISE 的原因是，I²C 总线的上升沿依赖于电阻上拉，需要的事件较长。若主模式释放 SCL 后立刻进行检测，则可能由于上升速度不够而检测到错误的从设备拉低。最终导致时钟高电平事件变长，总线时钟周期不稳定。

10.3.3.2 “开始”条件

设置“开始”命令后，进入主模式（SR2_MSL），并生成总线的开始条件。“开始”命令配置后，硬件不会检查总线是否忙；软件应检查总线忙状态（SR2_BUSY），在非忙状态下设置“开始”命令，并检查是否存在仲裁丢失状态（SR1_ARLO）出现。

当成功生成总线的开始条件后，开始位状态（SR1_SB）置起。清除开始位状态的方法是：读取到开始位后，将从设备地址写到数据寄存器中。

10.3.3.3 从设备地址发送

移位寄存器从数据寄存器中取出从设备地址，开始发送到 SDA 线上。在龙芯 1D100BA 中，只支持 7 位地址模式，地址阶段发送的一个字节中，高 7 位为地址，最低位为发送或接收模式的选择。最低位为低时，将进行主发送模式；最低位为高时，将进行主接收模式。

当发送的地址在应答周期接收到有效应答时，会产生地址状态位；若收到的为无应答，则会进行空闲状态等待新的命令（如再次开始或结束），此时 SCL 为低、SDA 为高。

与从模式类似，发送状态位（SR2_TRA）表示处于发送还是接收状态。

10.3.3.4 主发送模式

处理完地址状态位后，主设备通过移位寄存器将数据寄存器中的字节发送到 SDA 线上。

当需要发送数据，但发送数据寄存器没有有效的数时，会产生发送空状态（SR1_TXE）。若传输完一个数据字节后，仍出现发送空状态，则还会产生字节传输结束状态（SR1_BTTF）。即第一个需要发送的字节未写入时，只会有 TXE 状态，而后续字节未写入时，会有 BTTF 状态。

等待 TXE 或 BTTF 状态时，总线处于保持状态，即 SCL 为低，SDA 不变。

当需要结束传输时，写入“停止”命令（CR1_STOP）即可，主设备会在当前字节传输结束后发送停止条件。停止条件产生后，控制器自动返回从模式。

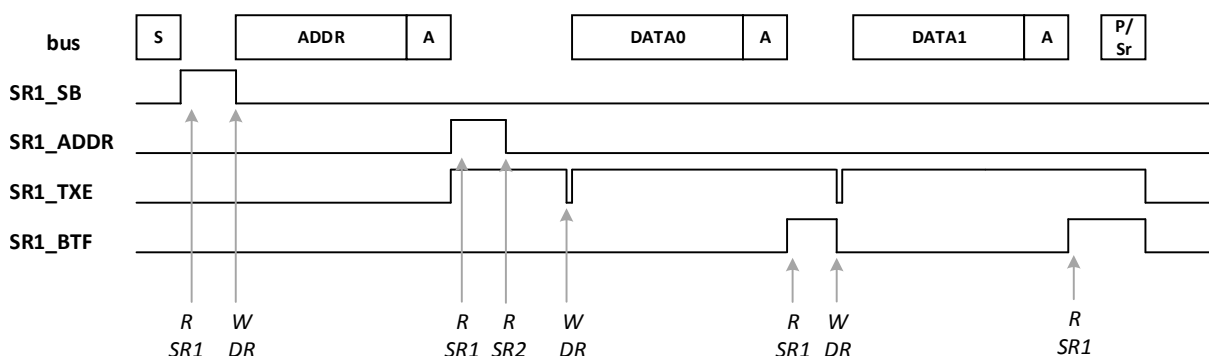


图 10.3: 主发送模式示意图

10.3.3.5 主接收模式

清除地址状态位后，从设备开始从 SDA 线接受数据放入移位寄存器。

收满一个字节后，若接收数据寄存器为空，则在应答周期完成后放入接收数据寄存器中，置起接收非空状态 (SR1_RXNE)，并继续进行下一个字节的接收。若应答配置 (CR1_ACK) 为无应答，则会结束本次传输，进入空闲状态等待下一个“开始”条件。

若接收数据寄存器不为空，则数据保留在移位寄存器中，并置起字节传输结束状态 (SR1_BTF) 并等待。在字节传输结束状态被清除前，控制器进入等待状态并拉低 SCL 线。

与从接收模式不同的是，主接收模式若需要结束传输，需要在最后一个字节的应答周期中给出无应答 (NACK) 条件。从设备收到无应答条件，释放总线，避免产生总线死锁现象。

发送完无应答条件后，控制器进入空闲状态，可发送再次开始或停止命令。开始或停止命令也可在最后一个字节传输过程中进行配置，传输完成后立刻执行。

为及时发送无应答条件，有下列注意事项：

- 能确保在收到状态后、下一个字节传输过程中完成寄存器配置时：
接收到倒数第二个字节后，通过清除 CR1_ACK 配置位的方式发送无应答条件。
若只需要传输一个字节，则在清除地址状态位后立刻清除应答配置位；事实上，由于应答配置位只对接收有效，在发送地址前后清除应答配置位均可达到相同目的。
- 无法确保及时完成寄存器配置时
这种条件下，需要在接收到倒数第三个字节后，就进行配置。当接收字节数大于 2 时，接收到倒数第三个字节后，不进行读出，直到倒数第二个字节也被接收在移位寄存器中；即此时接收数据寄存器中保存倒数第三个字节，移位寄存器中保存倒数第二个字节，此时 SR1_RXNE 和 SR1_BTF 状态位都置起；此时，清除应答配置位即可成功发送无应答条件。读取倒数第三个字节的同时，倒数第二个字节自动由移位寄存器进入接收数据寄存器，总线开始进行最后一个字节的传输。故在读取倒数第三个字节后配置开始或停止命令已经是安全的。
当接收字节数等于 2 时，情况有所不同。由于接收数据寄存器和移位寄存器共能接收 2 个字节，故需要特殊机制来保证最后一个字节无应答。此时，需要使用应答位置配置位 (CR1_POS) 来获得延迟应答，CR1_POS 为 0 时，延迟应答为 0；CR1_POS 为 1 时，延迟应答会在下一次接收应答时更新为应答配置位 (CR1_ACK) 的值。故在地址状态处理的前后位置，即可配置 CR1_POS 并清除 CR1_ACK，此时 2 个字节的应答会分别为有应答和无应答。接收到 SR1_BTF 状态时，表示 2 个字节都接收完成，读取数据寄存器两次即可。重新开始或停止命令在收到 SR1_BTF 状态后进行配置。
当接收字节数为 1 时，在清除地址状态前即清除应答配置位，以免错过配置时间。

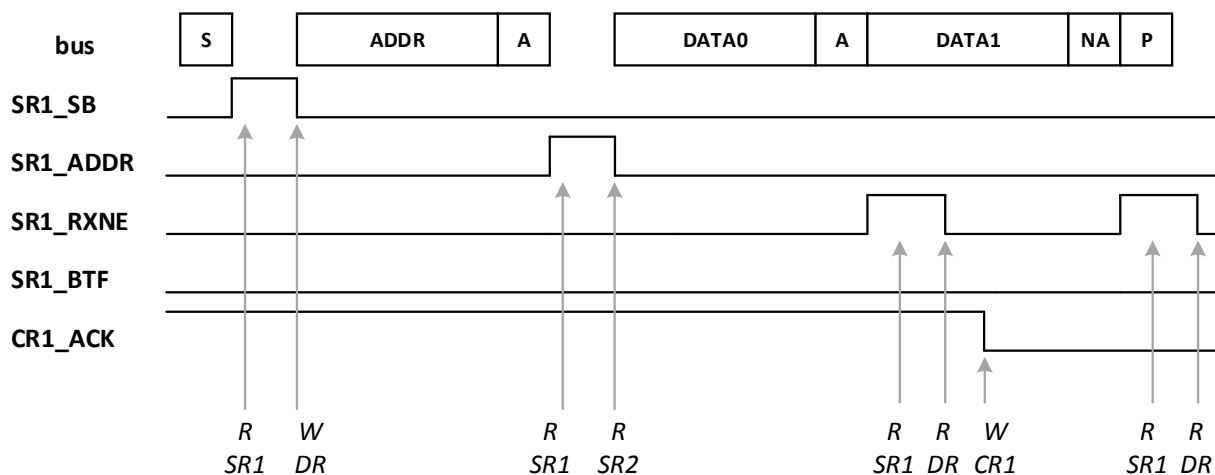


图 10.4: 主接收模式（及时）示意图

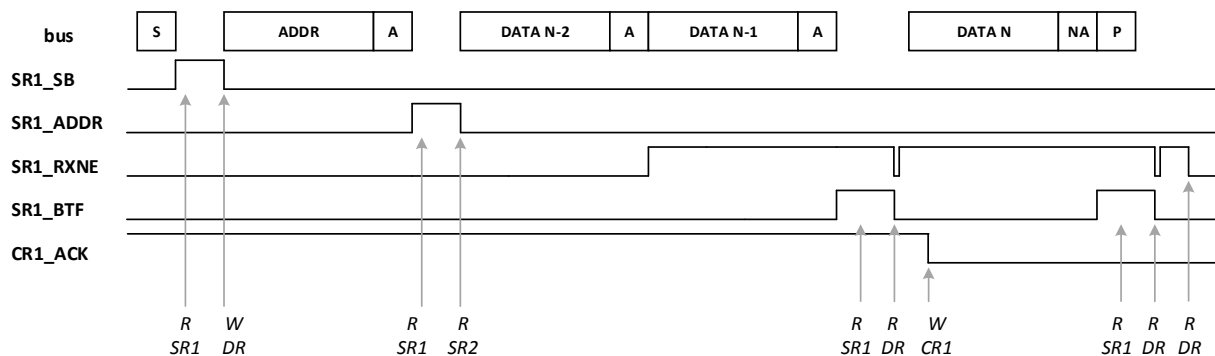


图 10.5: 主接收模式（非及时、3 字节）示意图

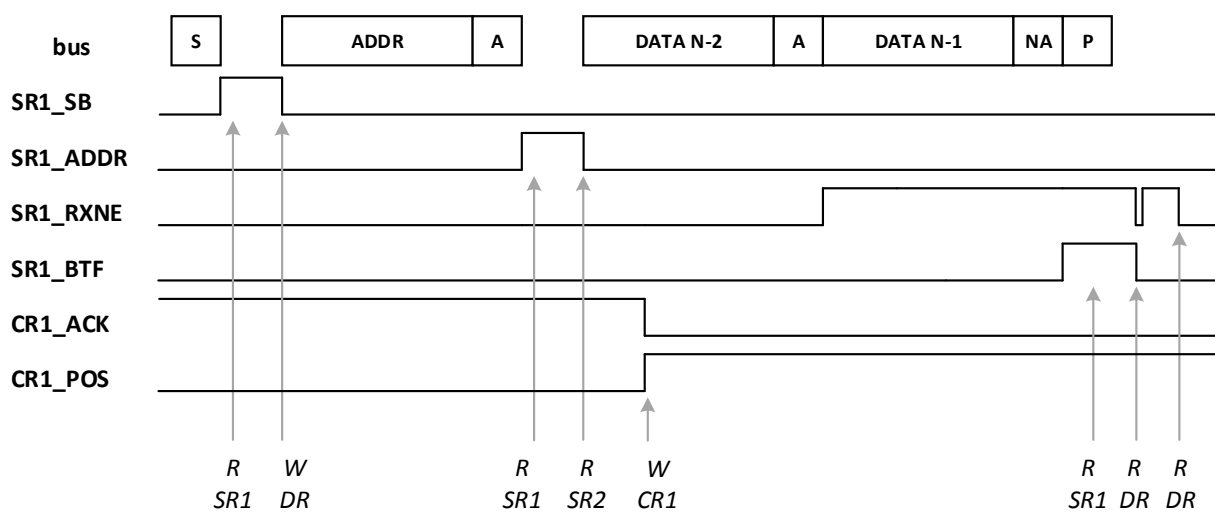


图 10.6: 主接收模式（非及时、2 字节）示意图

10.3.4 错误条件

下列为可能导致传输失败的错误条件。

- 总线错误（SR1_BERR）

当控制器处于地址或数据传输状态时，总线上出现了开始或停止条件。在从模式中，当前传输数据被丢弃，总线被释放；错误的开始条件会被视为正常的重新开始，等待地址或停止；错误的停止条件会被视为正常的停止。在主模式中，当前传输继续，软件可依据状态进行操作；若软件不中断当前传输，则后续有可能出现仲裁丢失而停止。

- 应答失败（SR1_AF）

产生应答失败时，表示发送的地址或数据未被接收。此时，内部状态回到空闲状态。从模式时，总线被释放，等待新的开始条件；主模式时，等待软件配置重新开始或停止命令。

- 仲裁丢失（SR1_ARLO）

当主模式欲发送的 SDA 为高，而 SDA 线为低时，产生仲裁丢失状态，表明其他主设备赢得总线控制权。此时，总线被释放，控制器回到从模式，进入空闲状态。即使获胜的主设备访问的是本设备，也无法进行应答，需等待下一个开始条件。

- 溢出错误（SR1_OVR）

溢出错误分为上溢和下溢两种，都是由于从模式下时钟伸展特性被禁用（CR1_NOSTRETCH）时，上溢由于数据寄存器非空时仍接收写入，下溢由于数据寄存器非满时仍读出发送。

当产生上溢时，最后一个接收的字节被丢弃；软件应清除非空状态，并以一定方法使发送方重发。当产生下溢时，数据寄存器的字节将被发送两次；用户应保证接收方能识别并无视该错误。当需要发送第一个字节时，必须在数据阶段的第一个 SCL 上升沿之前写入数据寄存器。

当与不支持时钟伸展的主设备进行传输时，用户必须自行定义合理的校验方式，来识别可能的溢出错误；并尽量保证及时对数据进行读出或写入。

10.4 中断

中断状态和对应的使能控制位如下表：

表 10.1: I²C 中断请求

中断事件	标记位	使能位
开始条件已发送（主模式）	SR1_SB	CR2_ITEVTEN
地址已发送（主模式）或地址命中（从模式）	SR1_ADDR	CR2_ITEVTEN
收到停止条件（从模式）	SR1_STOPF	CR2_ITEVTEN

中断事件	标记位	使能位
字节传输结束	SR1_STF	CR2_ITEVTEN
接收数据寄存器非空	SR1_RXNE	CR2_ITEVTEN and CR2_ITBUFEN
发送数据寄存器空	SR1_TXE	CR2_ITEVTEN and CR2_ITBUFEN
总线错误	SR1_BERR	CR2_ITERREN
仲裁丢失（主模式）	SR1_ARLO	CR2_ITERREN
应答失败	SR1_AF	CR2_ITERREN
溢出	SR1_OVR	CR2_ITERREN

10.5 寄存器定义

I²C 控制器的基址为 0x480。

表 10.2: I²C 控制器寄存器列表

名称	偏移	描述
I ² C_CR1	0x00	I ² C 控制寄存器 1
I ² C_CR2	0x04	I ² C 控制寄存器 2
I ² C_OAR	0x08	I ² C 从地址寄存器
I ² C_DR	0x10	I ² C 数据寄存器
I ² C_SR1	0x14	I ² C 状态寄存器 1
I ² C_SR2	0x18	I ² C 状态寄存器 2
I ² C_CCR	0x1c	I ² C 时钟控制寄存器
I ² C_TRISE	0x20	I ² C 上升时间寄存器

10.5.1 I²C 控制寄存器 1 (I²C_CR1)

偏移: 0x0

复位值: 32'h0

表 10.3: I²C 控制寄存器 1

位域	名称	访问	描述
31:16	reserved	–	保留
15	CR1_SWRST	RW	软复位 0: 未复位 1: 复位 该复位只复位内部控制逻辑, 不复位寄存器, 应配合 CR1_PE 使用 软件可配置和清除该位。
14	CR1_RECOVER	RW	总线恢复命令 发出 9 个 SCL 时钟和一个停止条件, 可用于解除死锁状态 (SCL 为高, 从设备将 SDA 拉低)。 软件可设置和清除该位, 硬件会在发出停止条件后或 CR1_PE 为 0 时清除该位。
13:12	reserved	–	保留

位域	名称	访问	描述
11	CR1_POS	RW	数据接收的应答位置 0: CR1_ACK 位控制当前移位寄存器正在接收的字节的应答 1: CR1_ACK 位控制将由移位寄存器接收的下一个字节的应答 硬件行为是在接收字节的应答周期完成时, 将 CR1_ACK 表示的值保存到寄存器中作为应答值; 该位无效时, 应答寄存器复位为有应答; 故开启时, 第一个应答为有应答, 下一个为在第一个应答周期保存的 CR1_ACK 表示的值。 软件可设置和清除该位, 硬件会在 CR1_PE 为 0 时清除该位。
10	CR1_ACK	RW	应答使能 0: 无应答返回 (在应答拍为高电平) 1: 收到一个字节后返回有应答 (地址命中或接收到数据后, 在应答拍为低电平) 软件可设置和清除该位, 硬件会在 CR1_PE 为 0 时清除该位。
9	CR1_STOP	RW	停止生成命令 0: 无命令 1: 主模式下在当前字节传输 (不含地址阶段) 或开始/恢复命令完成后, 生成总线停止条件; 从模式下在当前字节传输完成后 (不含地址阶段) 释放总线, 回到空闲状态。 停止命令在主接收模式未输出无应答条件时存在总线状态不确定的风险。 软件可设置和清除该位, 硬件会在总线停止条件后清除该位。
8	CR1_START	RW	开始生成命令 0: 无命令 1: 主模式下在当前字节传输 (不含地址阶段) 后, 生成重复开始条件; 从模式下在当前字节传输 (不含地址阶段) 后或空闲时, 生成开始条件并进入主模式。 开始命令无视总线状态, 软件应自行检查总线状态 (通过 SR2_BUSY) 并准备处理可能的仲裁丢失错误。 软件可设置和清除该位, 硬件会在总线开始条件后或 CR1_PE 为 0 时清除该位。
7	CR1_NOSTRETCH	RW	时钟伸展无效 (从模式) 0: 时钟伸展有效 1: 时钟伸展无效 软件可配置和清除该位。
6	CR1_ENGC	RW	广播呼叫使能 1: 使能广播呼叫, 地址 0x00 将得到应答 软件可配置和清除该位。
5:1	reserved	—	保留
0	CR1_PE	RW	控制器使能 1: 使能 在工作中清除该位会立刻重置控制器状态机, 无视总线 软件可配置和清除该位。

10.5.2 I²C 控制寄存器 2 (I²C_CR2)

偏 移: 0x4

复位值: 32'h0

本寄存器不含硬件置位或复位

表 10.4: I²C 控制寄存器 2

位域	名称	访问	描述
31:16	reserved	–	保留
15:13	CR2_FILTER	RW	数字滤波器配置 0: 不使用滤波器 1~ 7: 可滤除 1~ 7 个周期的脉冲
12:11	reserved	–	保留
10	CR2_ITBUFEN	RW	缓冲类中断使能 1: 使能缓冲类中断, 包括 TXE 和 RXNE
9	CR2_ITEVTEN	RW	事件类中断使能 1: 使能事件类中断, 包括 SB、ADDR、STOPF、BTF、TXE、RXNE
8	CR2_ITERREN	RW	错误类中断使能 1: 使能错误类中断, 包括 BERR、ARLO、AF、OVR
7:6	reserved	–	保留
5:0	CR2_FREQ	RW	设备时钟频率 表示 APB 接口时钟频率 (MHz), 用于生成数据建立和保持事件 (从模式)。在龙芯 1D100BA 中, 若使用外部 8M 晶体, 则该位域配置为 8; 若使用内部时钟, 该位域配置为 11 (内部 32M 时钟 3 分频)。

10.5.3 I²C 从地址寄存器 (I²C_OAR)

偏 移: 0x8

复位值: 32' h0

本寄存器不含硬件置位或复位

表 10.5: I²C 从地址寄存器

位域	名称	访问	描述
31:8	reserved	–	保留
7:1	OAR	RW	从设备地址 作为从模式地址阶段的 [7:1] 位进行地址命中的判断
0	reserved	–	保留

10.5.4 I²C 数据寄存器 (I²C_DR)

偏 移: 0x10

复位值: 32' h0

表 10.6: I²C 数据寄存器

位域	名称	访问	描述
31:8	reserved	–	保留
7:0	DR	RW	字节数据寄存器 写入时, 写到发送数据寄存器 (TXR) 中; 读出时, 读出接收数据寄存器 (RXR) 的值。从模式接收的地址不会放入 RXR 中。

10.5.5 I²C 状态寄存器 1 (I²C_SR1)

偏 移: 0x14

复位值: 32'h0

表 10.7: I²C 状态寄存器 1

位域	名称	访问	描述
31:12	reserved	—	保留
11	SRI_OVR	R	溢出状态位 详细说明见错误条件 软件向该位写 0 清除状态, 硬件在 CR1_PE 为 0 时清除该位。
10	SRI_AF	R	应答失败状态位 详细说明见错误条件 软件向该位写 0 清除状态, 硬件在 CR1_PE 为 0 时清除该位。
9	SRI_ARLO	R	仲裁丢失状态位 详细说明见错误条件 软件向该位写 0 清除状态, 硬件在 CR1_PE 为 0 时清除该位。
8	SRI_BERR	R	总线错误状态位 详细说明见错误条件 软件向该位写 0 清除状态, 硬件在 CR1_PE 为 0 时清除该位。
7	SRI_TXE	R	发送数据寄存器空 发送模式时数据寄存器为空。 在地址阶段不会置位, 收到无应答 (NACK) 条件时不会置位。 软件写数据寄存器后清除, 硬件在开始或停止条件后或在 CR1_PE 为 0 时清除。 该位在写数据寄存器清除后可能很快再次置起, 此时可再次写入。
6	SRI_RXNE	R	接收数据寄存器非空 接收模式时数据寄存器非空, 即收到了有效的数。 在地址阶段不会置位, 在出现仲裁丢失时不会置位。 软件写或读数据寄存器后清除, 硬件在 CR1_PE 为 0 时清除。 该位在读数据寄存器清除后可能很快再次置起, 因为移位寄存器中的数放到了接收数据寄存器, 此时可再次读出。
5	reserved	—	保留
4	SRI_STOPF	R	从模式停止位检测 从模式时, 在正常的应答后检测到总线停止条件后置位。 收到无应答 (NACK) 时不会置位。 要清除该位, 软件在读出该位为 1 后, 再写 I ² C_CR1 寄存器; 硬件在 CR1_PE 为 0 时清除。
3	reserved	—	保留
2	SRI_BTF	R	字节传输结束状态 当 CR1_NOSTRETCH 为 0 时可置位。接收模式时, 接收到一个字节且接收数据寄存器非空时置位; 发送模式时, 发送完一个数据字节后且发送数据寄存器为空时置位。 在收到无应答 (NACK) 条件时不会置位。 要清除该位, 软件在读出该位为 1 后, 再写或读数据寄存器; 硬件收到开始或停止状态, 或在 CR1_PE 为 0 时清除。

位域	名称	访问	描述
1	SR1_ADDR	R	地址阶段成功 主模式时发送地址成功，或从模式时接收地址命中。 收到 NACK 时不会置位。 要清除该位，软件在读出该位为 1 后，再读 I ² C_SR2 寄存器；硬件在 CR1_PE 为 0 时清除。
0	SR1_SB	R	开始条件成功 主模式成功生成开始条件后置位。 要清除改位，软件在读出该位为 1 后，再写数据寄存器（地址）；硬件在 CR1_PE 为 0 时清除。

10.5.6 I²C 状态寄存器 2 (I²C_SR2)

偏 移： 0x18

复位值： 32' h0

表 10.8: I²C 状态寄存器 2

位域	名称	访问	描述
31:5	reserved	–	保留
4	SR2_GENCALL	R	广播地址命中 当 CR1_ENGC 有效时，接收到了广播地址后置位。 硬件接收到开始或停止条件后清除，或在 CR1_PE 为 0 时清除。
3	reserved	–	保留
2	SR2_TRA	R	发送模式 0：接收模式 1：发送模式 当内部状态机为发送相关的状态时，该位为 1，否则均为 0。 当 CR1_PE 为 0，或各种会导致状态机离开发送状态的事件均会清除该位。
1	SR2_BUSY	R	总线忙 硬件检测到 SCL 或 SDA 线上为低时置位，检测到停止条件后清除。 该位在 CR1_PE 为 0 时仍会更新。
0	SR2_MSL	R	主从模式 0：从模式 1：主模式 默认为从模式，从模式下收到开始或恢复命令后变为主模式。 主模式接收到停止条件或仲裁丢失时回到从模式，在 CR1_PE 为 0 时为从模式。

10.5.7 I²C 时钟控制寄存器 (I²C_CCR)

偏 移： 0x1c

复位值： 32' h0

本寄存器不含硬件置位或复位

表 10.9: I²C 时钟控制寄存器

位域	名称	访问	描述
31:16	reserved	–	保留
15	CCR_F_S	RW	主模式选择 0: 标准模式 1: 快速模式
14	CCR_DUTY	RW	占空比控制 0: 快速模式 SCL 高低电平时间为 1:2 1: 快速模式 SCL 高低电平时间为 9:16
13:12	reserved	–	保留
11:0	CCR_CCR	RW	时钟分频控制 定义 SCL 的高电平时间 T_{high} 和低电平时间 T_{low} 。 标准模式时: $T_{high} = CCR_CCR \times T_{PCLK}, T_{low} = CCR_CCR \times T_{PCLK}$ 快速模式且 CCR_DUTY 为 0 时: $T_{high} = CCR_CCR \times T_{PCLK}, T_{low} = 2 \times CCR_CCR \times T_{PCLK}$ 快速模式且 CCR_DUTY 为 1 时: $T_{high} = 9 \times CCR_CCR \times T_{PCLK}, T_{low} = 16 \times CCR_CCR \times T_{PCLK}$ 例如, 要配置标准模式 100KHz 时钟, APB 时钟为 8MHz 时: $CCR_CCR = 5000ns/125ns = 40$

10.5.8 I²C 上升时间寄存器 (I²C_TRISE)

偏 移: 0x20

复位值: 32'h0

本寄存器不含硬件置位或复位

表 10.10: I²C 上升时间寄存器

位域	名称	访问	描述
31:6	reserved	–	保留
5:0	TRISE	RW	SCL 最大上升时间 (主模式) 根据总线协议规定的最大 SCL 上升时间与 APB 时钟进行配置。例如, 标准模式下最大上升事件为 1000ns, 则 TRISE 配置为 $TRISE = 1000ns/125ns + 1 = 9$ 。当除法结果不是整数时, 向下取整即可。 该位域用于在合适的时间判断是否出现时钟伸展, 从而有助于保持 SCL 频率稳定。

第十一章 SPI 控制器

11.1 概述

串行外围设备接口 SPI 总线技术是 Motorola 公司推出的多种微处理器、微控制器以及外围设备之间的一种全双工、同步、串行数据接口标准。

龙芯 1D100BA 中的 SPI 控制器仅作为主控端。对于软件而言，SPI 控制器除了有若干 IO 寄存器外还有一段映射到 SPI Flash 的只读 memory 空间。如果将这段 memory 空间分配在 0x1c000000，复位后不需要软件干预就可以直接访问，从而支持处理器从 SPI Flash 启动。SPI 的 IO 寄存器的基地址为 0x440，外部存储地址空间是 0x1e00,0000 ~ 0x1eff,ffff 共 16MB。

11.2 寄存器定义

SPI 控制器的基地址为 0x440。

表 11.1: SPI 控制器寄存器列表

名称	偏移	描述
SPCR	0x00	控制寄存器
SPSR	0x01	状态寄存器
DATA	0x02	数据寄存器
SPER	0x03	外部寄存器
PARAM	0x04	参数控制寄存器
SOFTCS	0x05	片选控制寄存器
TIMING	0x06	时序控制寄存器

11.2.1 控制寄存器 (SPCR)

偏 移： 0x00

复位值： 8'h12

表 11.2: 控制寄存器

位域	名称	访问	描述
7	spie	RW	中断数据使能 0: 关闭中断 1: 开启中断
6	spe	RW	系统工作使能 0: 停止 1: 启动
5: 4	reserved	-	保留

位域	名称	访问	描述
3	cpol	RW	时钟极性 表示无时钟时 SPI_CLK 的电平。 0: 低电平 1: 高电平
2	cpha	RW	时钟相位 0: 相位相同 1: 相位相反
1:0	spr	RW	时钟分频位 与 SPER 的 spre 位一起使用

11.2.2 状态寄存器 (SPSR)

偏 移: 0x01

复位值: 8' h05

表 11.3: 状态寄存器

位域	名称	访问	描述
7	spif	RW	中断标志位 0: 无中断 1: 有中断, 写 1 清零
6	wcol	RW	写寄存器溢出标志位 0: 无溢出 1: 写寄存器溢出, 写 1 清零
5	reserved	-	保留
4	busy	R	表示控制器忙 1: 表示 FIFO 非空或状态未完成
3	wffull	R	写寄存器满标志 1: 满
2	wfempty	R	写寄存器空标志 1: 空
1	rffull	R	读寄存器满标志 1: 满
0	rfempty	R	读寄存器空标志 1: 空

11.2.3 数据寄存器 (DATA)

偏 移: 0x02

复位值: 8' h00

表 11.4: 数据寄存器

位域	名称	访问	描述
7:0	data	RW	数据 写入则为发送数据, 读出则为接收数据

11.2.4 外部寄存器 (SPER)

偏 移: 0x03

复位值: 8' h00

表 11.5: 外部寄存器

位域	名称	访问	描述
7:6	icnt	RW	传输多少字节后发中断 0 ~ 3; 1 ~ 4
5:3	reserved	-	保留
2	mode	RW	接口模式 0: 采样与发送时机同时 1: 采样与发送时机错开半周期
1:0	spre	RW	时钟分频位 与 SPCR 的 spr 位一起设定分频比率

表 11.6: SPI 分频系数

spre,spr	分频系数
4' b0000	2
4' b0001	4
4' b0010	16
4' b0011	32
4' b0100	8
4' b0101	64
4' b0110	128
4' b0111	256
4' b1000	512
4' b1001	1024
4' b1010	2048
4' b1011	4096

11.2.5 参数控制寄存器 (PARAM)

偏 移: 0x04

复位值: 8' h03

表 11.7: 参数控制寄存器

位域	名称	访问	描述
7:4	clk_div	RW	时钟分频数选择 分频系数与 {spre,spr} 组合相同
3	dual_io	RW	双 IO 模式 优先级高于快速读
2	fast_read	RW	快速读模式

位域	名称	访问	描述
1	burst_en	RW	SPI flash 支持连续地址读模式
0	memory_en	RW	SPI flash 读使能 无效时 cs[0] 可由软件控制，需要使用简单模式时需将此位清零

11.2.6 片选控制寄存器 (SOFTCS)

偏 移: 0x05

复位值: 8' hf0

表 11.8: 片选控制寄存器

位域	名称	访问	描述
7:4	csn	RW	片选 对应使能有效时，控制 SPI 的四个片选信号，7:5 位受 SCS 控制，龙芯 1D100BA 只有片选 0 和 1。
3:0	csen	RW	片选使能 1: 对应片选生效

11.2.7 时序控制寄存器 (TIMING)

偏 移: 0x06

复位值: 8' h03

表 11.9: 时序控制寄存器

位域	名称	访问	描述
7:4	reserved	-	保留
3	SCS	RW	分离片选模式 0: 片选由 SOFTCS 直接控制 1: SOFTCS 选中的片选信号在需要发起 SPI 访问时才有效，并在访问结束后清除。
2	tFAST	RW	SPI flash 读采样模式 0: 上沿采样，间隔半个 SPI 周期 1: 上沿采样，间隔一个 SPI 周期
1:0	tCSH	RW	SPI flash 片选信号最短无效时间 以分频后时钟周期 T 计算。 0: 1T 1: 2T 2: 4T 3: 8T

11.3 接口时序

11.3.1 SPI 主控制器接口时序

接口时序如图11.1所示。

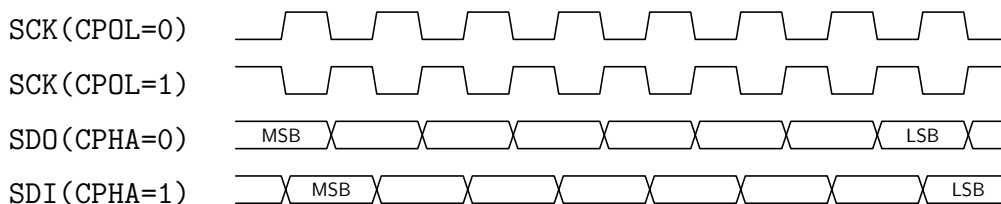


图 11.1: SPI 主控制器接口时序

11.3.2 SPI Flash 访问时序

SPI Flash 的访问时序如图11.2, 11.3, 11.4所示。

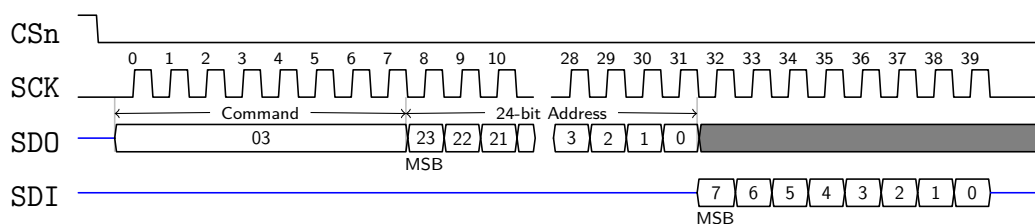


图 11.2: SPI Flash 标准读时序

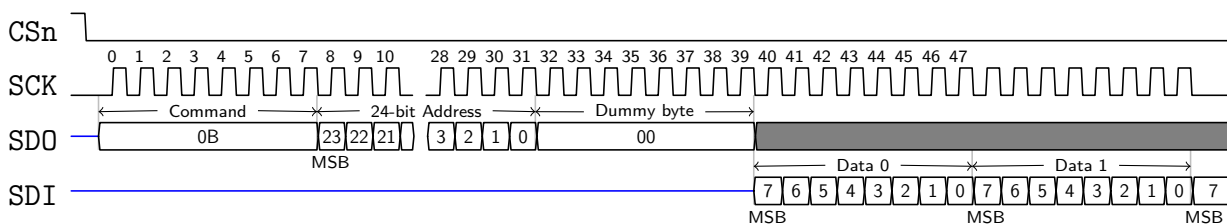


图 11.3: SPI Flash 快速读时序

11.4 使用指南

11.4.1 SPI 主控制器的读写操作

11.4.1.1 模块初始化

- 停止 SPI 控制器工作，对控制寄存器 `sper` 的 `spe` 位写 0
- 重置状态寄存器 `spsr`，对寄存器写入 8' b100_0000
- 设置外部寄存器 `sper`，包括中断申请条件 `sper[7:6]` 和分频系数 `sper[1:0]`，具体参考寄存器说明
- 配置 SPI 时序，包括 `sper` 的 `cpol`、`cpha` 和 `sper` 的 `mode` 位。`mode` 为 1 时是标准 SPI 实现，为 0 时为兼容模式。

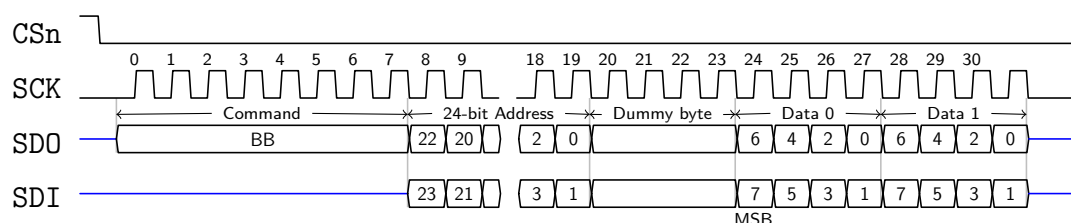


图 11.4: SPI Flash 双向 I/O 读时序

- 配置中断使能，sper 的 spie 位启动 SPI 控制器，对控制寄存器 sper 的 spe 位写 1

11.4.1.2 模块的发送/传输操作

- 往数据传输寄存器写入数据
- 传输完成后从数据传输寄存器读出数据。由于发送和接收同时进行，即使 SPI 从设备没有发送有效数据也必须进行读出操作。

11.4.1.3 中断处理

- 接收到中断申请
- 读状态寄存器 spsr 的值，若 spsr[2] 为 1 则表示数据发送完成，若 spsr[0] 为 1 则表示已经接收数据
- 读或写数据传输寄存器
- 往状态寄存器 spsr 的 spif 位写 1，清除控制器的中断申请

11.4.2 硬件 SPI Flash 读

11.4.2.1 初始化

- 将 SFC_PARAM 的 memory_en 位写 1。当 SPI 被选为启动设备时此位复位为 1。
- 设置读参数（时钟分频、连续地址读、快速读、双 I/O、tCSH 等）。这些参数复位值均为最保守的值。

11.4.2.2 更改参数

如果所使用的 SPI Flash 支持更高的频率或者提供增强功能，修改相应参数可以大大加快 Flash 的访问速度。参数的修改不需要关闭 SPI Flash 读使能 (memory_en)。具体参考寄存器说明。

11.4.3 混合访问 SPI Flash 和 SPI 主控制器

将 SPI Flash 读使能关闭后，软件就可直接控制 `csn[0]`，并通过 SPI 主控制器访问 SPI 总线。这意味着在进行此操作时，不能从 SPI Flash 中取指。

除了读以外，SPI Flash 还实现了很多命令（如擦除、写入），具体参见相关 Flash 的文档。

此页留空

第十二章 UART 控制器

12.1 概述

龙芯 1D100BA 集成了 1 个 UART 控制器，使用总线接口时钟作为波特率时钟，提供与其他外部设备串行通信的功能，例如与一台计算机以 RS232 为标准使用串行线路进行通信。UART 控制器为两线模式，不支持 MODEM 信号。

12.2 寄存器定义

UART 寄存器基地址为 0x460。

表 12.1: UART 寄存器列表

名称	偏移	描述
DAT	0x00	数据寄存器 (LCR[7]=0)
IER	0x01	中断使能寄存器 (LCR[7]=0)
IIR	0x02	中断状态寄存器 (LCR[7]=0, 只读)
FCR	0x02	FIFO 控制寄存器 (LCR[7]=0, 只写)
LCR	0x03	线路控制寄存器
MCR	0x04	MODEM 控制寄存器
LSR	0x05	线路状态寄存器
MSR	0x06	MODEM 状态寄存器
DL_L	0x00	分频值低字节寄存器 (LCR[7]=1)
DL_H	0x01	分频值高字节寄存器 (LCR[7]=1)
DL_D	0x02	分频值小数寄存器 (LCR[7]=1)
SPC	0x04	采样点配置寄存器 (LCR[7]=1)

12.2.1 数据寄存器 (DAT)

偏 移: 0x00

复位值: 8'h0

表 12.2: 数据寄存器

位域	名称	访问	描述
7:0	data	RW	数据 读此寄存器时为收到的数据，写此寄存器将待发送的数据写入发送 FIFO

12.2.2 中断使能寄存器 (IER)

偏 移: 0x01

复位值: 8' h0

表 12.3: 中断使能寄存器

位域	名称	访问	描述
7:3	reserved	–	保留
2	ILE	RW	线路状态中断使能 值为 1 使能
1	ITE	RW	发送状态中断使能 值为 1 使能
0	IRE	RW	接收状态中断使能 值为 1 使能

12.2.3 中断状态寄存器 (IIR)

偏 移: 0x02

复位值: 8' h0

表 12.4: 中断状态寄存器

位域	名称	访问	描述
7:4	reserved	–	保留
3:1	II	RO	中断源 中断源, 详见下表。
0	INTPn	RO	中断未决状态 低有效, 表示存在未处理的中断。

表 12.5: 中断控制器功能表

II	优先级	中断类型	中断源	中断复位控制
3'b011	1	线路状态	接收 fifo 溢出、奇偶校验错误、帧错误 (停止符异常) 或打断中断 (接受到整帧长内都为 0)	读 LSR
3'b010	2	接收状态	接收 fifo 内数据数量达到了 trigger 值	读数据寄存器至少 于 trigger 值
3'b110	2	接收状态	接收超时, 接收 fifo 中有字符数据且在后续 4 帧时间内无操作	读数据寄存器
3'b001	3	发送状态	发送 FIFO 为空	读 IIR 或写数据寄存器
3'b000	4	MODEM 状态	MODEM 状态发生改变	读 MODEM 状态寄存器

12.2.4 FIFO 控制寄存器 (FCR)

偏 移: 0x02

复位值: 8' h0

表 12.6: FIFO 控制寄存器

位域	名称	访问	描述
7:6	trigger	WO	接收中断状态所需 trigger 单位为字节 0x0: 1 字节 0x1: 2 字节 0x2: 3 字节 0x3: 4 字节
5:4	reserved	–	保留
3	txbrk	WO	发送打断帧 写 1 后，在发送队列中加入打断帧。
2	txrst	WO	复位发送 FIFO 写 1 后，txfifo 复位。
1	rxrst	WO	复位接收 FIFO 写 1 后，rxfifo 复位。
0	reserved	–	保留

12.2.5 线路控制寄存器 (LCR)

偏 移: 0x03

复位值: 8'h3

表 12.7: 线路控制寄存器

位域	名称	访问	描述
7	dlab	RW	分频器模式 0: 访问正常寄存器; 1: 访问分频值寄存器。
6	bcb	RW	打断控制位 0: 正常操作; 1: 串口输出置为 0 (打断状态)。
5	spd	RW	指定奇偶校验位 0: 使用传统奇偶校验, 见 eps(LCR[4]) ; 1: 使用指定奇偶校验, 校验位始终为 !eps(LCR[4])。
4	eps	RW	奇偶校验位选择 0: 奇校验 (即发送数据中, 包括校验位在内, 有奇数个 1); 1: 偶校验 (即发送数据中, 包括校验位在内, 有偶数个 1)。
3	pe	RW	奇偶校验位使能 0: 无奇偶校验位; 1: 使能, 输出校验位, 输入判断校验位。
2	sb	RW	生成停止位位数 0: 1 个停止位; 1: bec 为 5 时 1.5 个停止位, 其他值时 2 个停止位。
1:0	bec	RW	字符位数 0: 5 位; 1: 6 位; 2: 7 位; 3: 8 位。

12.2.6 MODEM 控制寄存器 (MCR)

偏 移: 0x4

复位值: 8'h0

表 12.8: MODEM 控制寄存器

位域	名称	访问	描述
7:5	reserved	–	保留
4	loopback	RW	自回环模式控制位 0: 正常工作模式 1: 自回环模式: TX–RX, RTSn–CTS _n , DTRn–DSR _n , OUT _{1n} –DCD _n , OUT _{2n} –RI _n
3:0	reserved	–	保留

在自回环模式下, 各输入信号的值在内部直接由对应的输出信号决定, 而输出信号则对外输出高电平。

12.2.7 线路状态寄存器 (LSR)

偏 移: 0x05

复位值: 8'h0

对此寄存器进行读操作时, LSR[4:1] 和 LSR[7] 被清零, LSR[6:5] 在给传输 FIFO 写数据时清零, LSR[0] 则对接收 FIFO 进行判断。

表 12.9: 线路状态寄存器

位域	名称	访问	描述
7	error	RO	错误表示位 0: 无错误 1: 有奇偶校验错误、帧错误或打断中断
6	TE	RO	传输为空表示位 0: 有数据 1: 传输 FIFO 和传输移位寄存器都为空。给传输 FIFO 写数据时清零
5	TFE	RO	传输 FIFO 为空表示位 0: 有数据 1: 当前传输 FIFO 为空, 给传输 FIFO 写数据时清零
4	BI	RO	打断中断表示位 0: 没有中断 1: 接收到起始位+数据+奇偶位+停止位都是 0, 即有打断中断
3	FE	RO	帧错误表示位 0: 没有错误 1: 接收的数据没有停止位
2	PE	RO	奇偶校验位错误表示位 0: 没有奇偶错误 1: 当前接收数据有奇偶错误

位域	名称	访问	描述
1	OE	RO	数据溢出表示位 0: 无溢出 1: 有数据溢出
0	DR	RO	接收数据有效表示位 0: 在 FIFO 中无数据 1: 在 FIFO 中有数据

12.2.8 MODEM 状态寄存器 (MSR)

偏 移: 0x06

复位值: 8' h0

两线模式无 Modem 状态, 请保持相关寄存器为默认值。

表 12.10: MODEM 状态寄存器

位域	名称	访问	描述
7:0	reserved	-	保留

12.2.9 分频值低字节寄存器 (DL_L)

偏 移: 0x00

复位值: 8' h0

表 12.11: 分频值低字节寄存器

位域	名称	访问	描述
7:0	low	RW	分频值整数部分低字节

12.2.10 分频值高字节寄存器 (DL_H)

偏 移: 0x01

复位值: 8' h0

表 12.12: 分频值高字节寄存器

位域	名称	访问	描述
7:0	high	RW	分频值整数部分高字节

12.2.11 分频值小数寄存器 (DL_D)

偏 移: 0x02

复位值: 8' h0

表 12.13: 分频值小数寄存器

位域	名称	访问	描述
7:0	dec1	RW	分频值小数部分的二进制表示，如 0xc0 表示 0.75

12.2.12 采样点配置寄存器 (SPC)

偏 移: 0x04

复位值: 8'h77

表 12.14: 采样点配置寄存器

位域	名称	访问	描述
7	reserved	—	保留
6:4	ws1	RW	采样窗口大小 1 每 bit 取值点离结束点间的采样间隔数目 -1。
2:0	ws0	RW	采样窗口大小 0 每 bit 起始点到取值点间的采样间隔数目 -1。

由 DL_H、DL_L、DL_D 组成分频值寄存器 DL，则波特率为 $\frac{f_{APB}}{win_size \times DL}$ ，采样点总数 $win_size = ws0 + ws1 + 2$ ，默认为 16。假设 UART 控制器的总线收发时钟的频率为 10667 kHz。如果目标波特率为 115200，则 $DL = 5.787037$ ，故 $DL_H = 0x0$ ， $DL_L = 0x5$ ， $DL_D = 0xca$ 。波特率的配置值与期望值的误差应在 3% 以内，否则无法识别所有数据位，将导致乱码。

12.3 使用指南

12.3.1 配置流程

假设我们的 UART 总线收发时钟为 10667kHz，目标波特率为 115200，我们需要执行下列初始化操作：

1. 打开分频器

- 置 LCR 的 dlab 位为 1
- 置 DL_D 的值为 0xca
- 置 DL_H 的值为 0x0
- 置 DL_L 的值为 0x5
- 置 LCR 的 dlab 位为 0

2. 配置 LCR

- 根据帧的格式配置 LCR 的值

3. 配置 FCR

- 置 FCR 为 0xc6，RX 接收的 trigger 值为 4，同时复位发送和接收数据通路

4. 开中断使能

- 置 IER 为 0x1, 打开接收通路中断使能

此页留空

第十三章 MBUS 控制器

13.1 概述

龙芯 1D100BA 的 MBUS 控制器使用 32K 时钟进行数据收发, 支持 2400 波特率收发数据和帧过滤功能。经过配置后支持对 CJ/T188-2004 标准的帧进行过滤。

13.2 帧过滤

CJ/T188-2004 标准的帧是 MBUS 控制器的主要处理目标。表13.1为 CJ/T188-2004 标准的帧的结构定义。

表 13.1: CJ/T188-2004 标准帧结构

字节号	Code	描述
	FEh	2~ 4 个前导符。 前导符的值为 FEh
0	68h	MBUS 长帧开始标志
1	T	表的类型代码
2~ 8	A0~ A6	表计地址 (全为 AAh 则为广播地址)
9	CTR	控制码
10	L	数据域长度 (L 到 CS 间的字节数), L 的值最小为 3 应答帧的 L 小于或等于 64h, 请求帧的 L 小于或等于 32h
11~ 10+L	DATA	L 字节数据域 (DATA)。 包括数据标识、序列号和数据, 其结构随控制码的功能改变
11+L	CS	校验码 从帧起始符开始到校验码之前的所有各字节进行二进制算术累加, 不计超过 FFh 的溢出值
12+L	16h	帧结束标志

以 CJ/T188-2004 标准的帧为例, 一个帧由: 若干个前导符、帧的类型标识、设备类型标识、设备地址、帧数据、帧校验码、帧结束标识构成。MBUS 控制器的帧过滤机制, 通过比较帧前导符后的“帧的类型标识”、“设备类型标识”、“表计地址”来判断收到的帧是否是发送给自己所属的“表设备”。当匹配成功后, MBUS 控制器将帧中除前导字符外的内容送入软件可见 (从 DAT 端口读取) 的接收缓冲区中。为了实现帧过滤功能, MBUS 控制器在 UART 控制器的基础上在接收缓冲区前插入一个帧检测的控制逻辑和一个环形缓冲区。当帧检测功能被关闭时, MBUS 控制器保持 UART 控制

器的处理流程，环形缓冲区和存储帧匹配数据的触发器作为接收数据进入接收缓冲区前的数据缓冲使用。当帧检测功能被使能后，原来送入接收缓冲区的字符由帧检测逻辑进行下列处理：

1. 检测帧前导字符

在检测前导字符时，使用 2 个参数：前导字符值、最小次数。当接收到的连续的前导字符数不小于最小次数时，认为可以开始帧内容的检测，并保存当前循环缓冲区写指针的地址。前导字符不存入环形缓冲区（16 字节）。满足最小前导字符数后的第一个非前导字符字节被存入环形缓冲区。

2. 帧内容匹配

每收到一个新字节就进行内容匹配。收到的新字节顺序存入循环缓冲区。帧匹配的时候使用 7 个参数：匹配字节数（最多 16 字节）、16 个匹配字节值、16 位匹配字节掩码（设置该字节是否进行匹配）、广播地址起始位置（在进行匹配的 16 字节中的位置）、广播地址长度、广播地址值（8 字节）、8 位广播地址掩码。当接收到的内容满足匹配要求时，产生中断，同时打开将环形缓冲区内容导入接收缓冲区中的使能。

3. 帧结束检测

帧结束的检测使用超时机制，在帧前导字符后在规定的时间内没有新的字符进入就认为当前的帧结束，帧检测状态机重回前导字符检测状态。如果直到帧结束也未能和配置的地址匹配进行匹配，循环缓冲区的写指针恢复成帧检测前导后存储的写指针的值。

13.3 寄存器定义

MBUS 寄存器基地址为 0x300 。

表 13.2: MBUS 寄存器列表

名称	偏移	描述
DAT/DL_L	0x00	数据寄存器/分频值低字节寄存器
IER/DL_H	0x01	中断使能寄存器/分频值高字节寄存器
IIR	0x02	中断状态寄存器
FCR/DL_D	0x02	FIFO 控制寄存器/分频值小数寄存器
LCR	0x03	线路控制寄存器
sample_ctrl	0x04	bit 窗口划分和采样控制寄存器
LSR	0x05	线路状态寄存器
TF_CNT	0x06	发送队列数据存量
STATUS	0x07	状态寄存器
FSM_CTRL0	0x08	帧过滤控制寄存器 0
PRE_CHAR	0x09	帧过滤前导字符值前导字符值
FSM_CTRL1	0x0a	帧过滤控制寄存器 1

名称	偏移	描述
BR_MASK	0x0b	帧过滤广播地址掩码
FSM_MASKL	0x0c	帧过滤掩码低 8 位
FSM_MASKH	0x0d	帧过滤掩码高 8 位
BR_CTRL	0x0e	帧过滤广播地址参数
BUF_CNT	0x0f	帧过滤环形缓冲区数据量
FSM_TABLE	0x10 - 0x1f	帧过滤比较字节表
BR_TABLE	0x20 - 0x27	帧过滤广播地址表
FTIMEOUTL	0x28	帧超时计数比较的低 8 位值
FTIMEOUTH	0x29	帧超时计数比较高 8 位值
BUF_RPTR	0x2a	帧过滤环形缓冲区的读指针
Rev	0x2b - 0x2f	保留
FSM_BUF	0x30 - 0x3f	帧过滤环形缓冲区数据直接读取空间

13.3.1 数据寄存器 (DAT)

偏 移: 0x00

复位值: 8' h0

表 13.3: 数据寄存器

位域	名称	访问	描述
7:0	data	RW	数据 读此寄存器时为收到的数据, 写此寄存器将待发送的数据写入发送 FIFO

13.3.2 中断使能寄存器 (IER)

偏 移: 0x01

复位值: 8' h00

表 13.4: 中断使能寄存器

位域	名称	访问	描述
7:6	reserved	-	保留
5	FSM_DONE _INT_EN	RW	FSM 帧结束中断使能 值为 1 使能
4	FSM_MATCH _INT_EN	RW	FSM 帧匹配中断使能 值为 1 使能
3	IME	RW	Modem 状态中断使能 值为 1 使能
2	ILE	RW	线路状态中断使能 值为 1 使能
1	ITE	RW	发送状态中断使能 值为 1 使能
0	IRE	RW	接收状态中断使能 值为 1 使能

13.3.3 中断状态寄存器 (IIR)

偏 移: 0x02

复位值: 8' h01

表 13.5: 中断状态寄存器

位域	名称	访问	描述
7:6	reserved	–	保留
5	FSM_DONE	RO	中断未决状态 高有效, 表示当前符合过滤条件的帧接收完毕
4	FSM_MATCH	RO	帧匹配中断标志位 高有效, 表示出现符合过滤条件的帧
3:1	II	RO	中断源 中断源, 详见下表。
0	INTPn	RO	中断未决状态 低有效, 表示存在未处理的中断

表 13.6: 中断控制器功能表

II	优先级	中断类型	中断源	中断复位控制
3' b011	1	线路状态	奇偶、溢出或帧错误, 或打断中断	读 LSR
3' b010	2	接收状态	接收到的数据数量达到了 trigger 值	读数据寄存器
3' b110	2	接收状态	接收超时, 接收缓冲中有字符数据且在后续两个字符时间内无操作	读数据寄存器
3' b001	3	发送状态	发送 FIFO 为空	写数据寄存器或读中断状态寄存器
3' b000	4	Modem 状态	reserved, 两线串口实现无此中断源	读 Modem 状态寄存器

13.3.4 FIFO 控制寄存器 (FCR)

偏 移: 0x02

复位值: 8' h80

表 13.7: FIFO 控制寄存器

位域	名称	访问	描述
7:3	trigger	WO	接收中断状态所需 trigger 单位为字节 0x0, 0x1: 1 字节 0x2: 2 字节 ... 0x10: 16 字节 其它值: 保留
2	txreset	WO	复位发送 FIFO 此位置位后自动清零
1	rxreset	WO	复位接收 FIFO 此位置位后自动清零

位域	名称	访问	描述
0	reserved	-	保留

13.3.5 线路控制寄存器 (LCR)

偏 移: 0x03

复位值: 8'h03

表 13.8: 线路控制寄存器

位域	名称	访问	描述
7	dlab	RW	分频器模式 0: 访问正常寄存器 1: 访问分频值寄存器
6	becb	RW	打断控制位 0: 正常操作 1: 串口输出置为 0 (打断状态)
5	spd	RW	指定奇偶校验位 0: 不用指定奇偶校验位 1: 如果 eps 为 1 则传输和检查奇偶校验位为 0; 如果 eps 为 0 则传输和奇偶校验位为 1
4	eps	RW	奇偶校验位选择 0: 奇校验 1: 偶校验
3	pe	RW	奇偶校验位使能 0: 无奇偶校验位 1: 使能, 输出校验位, 输入判断校验位
2	sb	RW	生成停止位位数 0: 1 个停止位 1: bec 为 5 时 1.5 个停止位, 其他值时 2 个停止位
1:0	bec	RW	字符位数 0: 5 位 1: 6 位 2: 7 位 3: 8 位

13.3.6 bit 窗口划分和采样控制寄存器 (sample_ctrl)

偏 移: 0x04

复位值: 8'h38

表 13.9: bit 窗口划分和采样控制寄存器

位域	名称	访问	描述
7:4	sample_point	RW	bit 采样点位置 bit 采样点位于 bit 划分窗口中的位置 其允许值为 win_size -1 到 1 当 win_size 为 0 时, 其值的允许范围为: 15 ~ 1
3:0	win_size	RW	bit 采样窗口长度 将 1bit 的时间划分为多少份 0x0: 16 份 0x1, 0x2: 保留值 0x3: 3 份 ... 0xf: 15 份

13.3.7 线路状态寄存器 (LSR)

偏 移: 0x05

复位值: 8' h00

对此寄存器进行读操作时, LSR[4:1] 和 LSR[7] 被清零, LSR[6:5] 在给传输 FIFO 写数据时清零, LSR[0] 则对接收 FIFO 进行判断。

表 13.10: 线路状态寄存器

位域	名称	访问	描述
7	error	RO	错误表示位 0: 无错误 1: 有奇偶校验错误、帧错误或打断中断
6	TE	RO	传输为空表示位 0: 有数据 1: 传输 FIFO 和传输移位寄存器都为空。给传输 FIFO 写数据时清零
5	TFE	RO	传输 FIFO 为空表示位 0: 有数据 1: 当前传输 FIFO 为空, 给传输 FIFO 写数据时清零
4	BI	RO	打断中断表示位 0: 没有中断 1: 接收到起始位+数据+奇偶位+停止位都是 0, 即有打断中断
3	FE	RO	帧错误表示位 0: 没有错误 1: 接收的数据没有停止位
2	PE	RO	奇偶校验位错误表示位 0: 没有奇偶错误 1: 当前接收数据有奇偶错误
1	OE	RO	数据溢出表示位 0: 无溢出 1: 有数据溢出
0	DR	RO	接收数据有效表示位 0: 在 FIFO 中无数据 1: 在 FIFO 中有数据

13.3.8 分频值低字节寄存器 (DL_L)

偏 移: 0x00

复位值: 8' h0

表 13.11: 分频值低字节寄存器

位域	名称	访问	描述
7:0	low	RW	分频值整数部分低字节

13.3.9 分频值高字节寄存器 (DL_H)

偏 移: 0x01

复位值: 8' h0

表 13.12: 分频值高字节寄存器

位域	名称	访问	描述
7:0	high	RW	分频值整数部分高字节

13.3.10 分频值小数寄存器 (DL_D)

偏 移: 0x02

复位值: 8' h0

表 13.13: 分频值小数寄存器

位域	名称	访问	描述
7:0	deci	RW	分频值小数部分的二进制表示, 如 0xc0 表示 0.75

由 DL_H、DL_L、DL_D 组成分频值寄存器 DL, 则波特率为 $\frac{32768}{8 \times DL}$ 。MBUS 控制器的总线收发时钟的频率为 32768 Hz。如果目标波特率为 2400, 则 $DL = 1.7046875$, 故 $DL_H = 0x0$, $DL_L = 0x1$, $DL_D = 0xb6$ 。波特率的配置值与期望值的误差应在 3% 以内, 否则无法识别所有数据位, 将导致乱码。

13.3.11 发送队列中待发送的数据量 (TF_CNT)

偏 移: 0x06

复位值: 8' h00

表 13.14: 发送队列中待发送的数据量

位域	名称	访问	描述
7	loopback	RW	自回环模式控制位 0: 正常工作模式; 1: 自回环模式
6	mbus_clk_sel	RW	MBUS 收发时钟选择位 0: 使用内部 32K 时钟作为 MBUS 的收发时钟; 1: 使用内部总线时钟作为 MBUS 的收发时钟
5	reserved	—	保留
4:0	Tf_count	RO	发送队列中待发送的数据量 单位为字节

13.3.12 状态寄存器 (STATUS)

偏 移: 0x07

复位值: 8' h00

表 13.15: 状态寄存器

位域	名称	访问	描述
7	RX_RST	RO	接收数据通路中 32K 时钟域的复位状态 0: 不在复位状态; 1: 在复位状态
6	CLK32K_RST	RO	控制逻辑 32K 时钟域的复位状态 0: 不在复位状态; 1: 在复位状态
5	flush_wait	RO	接收数据通路中数据丢弃等待标识 此位为 1, 表示 RX 数据通路正处于 flush 数据的过程中
4:0	Rf_count	RO	接收队列中的数据量 单位为字节

13.3.13 帧过滤控制寄存器 0 (FSM_CTRL 0)

偏 移: 0x08

复位值: 8'h00

表 13.16: 帧过滤控制寄存器 0

位域	名称	访问	描述
7	reserved	—	保留
6	Debug_mode_sync	RO	Debug_mode 观测位 0 此位为 1 表示 debug_mode 的配置已经生效。当此位由 1 变为 0, 表示从 debug_mode 恢复为正常工作模式
5	Debug_mode	RW	调试模式控制位 0: 正常工作模式; 1: 调试环模式
4	do_flush	RW	RX 数据通路数据丢弃控制位 向此位写入 1, 执行刷 RX 数据通路的操作 (丢弃已经收到的帧数据直到当前正在接收的帧数据结束为止)。数据丢弃操作完成后, 此位自动清零
3	fsm_cfg_en	RW	帧过滤参数配置使能 0: 正常工作模式; 1: 帧过滤参数配置模式
2	fsm_work_en	RW	帧过滤工作使能
1	fsm_done	RWIC	帧接收的完成标识 写入 1 清零, 当 do_flush 置 1 时, 此位清零
0	fsm_match	RWIC	帧过滤匹配标识 此位为 1 表示: 接收到匹配的帧。写入 1 清零, 当 do_flush 置 1 时, 此位清零

在自回环模式下, 原 TX 的输出被直接在内部引入 RX 的输入, TX 对外输出的管脚输出高电平。在调试模式下, 软件可以通过偏移从 0x30 起始的地址直接读取帧过滤缓冲区中的内容。do_flush 置位后, 帧检测状态机强制进入等待帧结束的状态。只有在 fsm_cfg_en 为 1 时, 才可以配置地址偏移从 0x10 到 0x1f 的 FSM_TABLE。

表 13.17: 帧过滤工作模式

fsm_cfg_en	fsm_work_en	工作模式	描述
0	0	帧过滤机制关闭模式	在此模式下， fsm_table 和帧过滤 buf 均被用于存储接收到的数据。接收到的数据在经过 FSM 中 buf 的缓冲后直接送入串口控制器的数据接收 FIFO 中。
0	1	帧过滤机制开启模式	在此模式下，帧过滤机制工作。接收到的数据在帧前导字符结束后，先被存入帧过滤 buf 并进行帧过滤匹配。如果帧过滤机制匹配成功，帧过滤 buf 和后续收到的帧数据被存入数据接收 FIFO；如果帧过滤机制匹配不成功，当前帧被存入帧过滤 buf 中的数据被丢弃
1	0	帧过滤参数配置模式	用于配置 FSM_TABLE 的模式
1	1	保留模式	不允许出现的工作模式

13.3.14 帧过滤前导字符值 (PRE_CHAR)

偏 移: 0x09

复位值: 8' h00

表 13.18: 帧过滤前导字符值

位域	名称	访问	描述
7:0	PRE_CHAR	RW	帧前导字符的值

13.3.15 帧过滤控制寄存器 1 (FSM_CTRL 1)

偏 移: 0x0a

复位值: 8' h00

表 13.19: 帧过滤控制寄存器 1

位域	名称	访问	描述
7:4	pre_char_min_cnt	RW	帧前导字符的最小数量 当连续接收到的前导字符达到此参数值后，第一个非前导字符的数据被认为是帧的第一个字节。当此值为零时，帧没有前导字符
3:0	fsm_cnt	RW	参与帧过滤的字节数 从帧的第一个非前导字符开始，参与帧过滤比较的字节数。 0: 1 字节 1: 2 字节 2: 3 字节 ... f: 16 字节

13.3.16 帧过滤广播地址掩码 (BR_MASK)

偏 移: 0x0b

复位值: 8' h00

表 13.20: 帧过滤广播地址掩码

位域	名称	访问	描述
7:0	BR_MASK	RW	参与帧过滤的对应广播地址的掩码 0: 参与比较 1: 不参与比较

BR_MASK[N] 为 0, 则 br_addrN 参与广播地址的比较; BR_MASK[N] 为 1, 则 br_addrN 不参与广播地址的比较。N 的值为 0 - 7。Br_addr0 与最先收到广播地址进行比较; br_addr7 与最后收到的广播地址进行比较 (此时, pre_char_min_cnt 为 8)。

13.3.17 帧过滤掩码低 8 位 (FSM_MASKL)

偏 移: 0x0c

复位值: 8' h00

表 13.21: 帧过滤掩码低 8 位

位域	名称	访问	描述
7:0	FSM_MASKL	RW	Fsm_table 0 - 7 参与帧过滤的字节掩码 0: 参与比较 1: 不参与比较

13.3.18 帧过滤掩码高 8 位 (FSM_MASKH)

偏 移: 0x0d

复位值: 8' h00

表 13.22: 帧过滤掩码高 8 位

位域	名称	访问	描述
7:0	FSM_MASKH	RW	Fsm_table 8 - 15 参与帧过滤的字节掩码 0: 参与比较 1: 不参与比较

13.3.19 帧过滤广播地址参数 (BR_CTRL)

偏 移: 0x0e

复位值: 8' h00

表 13.23: 帧过滤广播地址参数

位域	名称	访问	描述
7:4	Br_cnt	RW	广播地址长度 允许值为 0 – 8 当 Br_cnt 为 0 时不进行广播地址的比较
3:0	Br_pos	RW	广播地址在帧中起始的位置 0x0: 帧在前导字符后的第 1 个字符 0x1: 帧在前导字符后的第 2 个字符 ... 0xf: 帧在前导字符后的第 16 个字符

13.3.20 帧过滤缓冲区中的有效数据个数 (BUF_COUNT)

偏 移: 0x0f

复位值: 8' h00

表 13.24: 帧过滤缓冲区中的有效数据个数

位域	名称	访问	描述
7:0	BUF_COUNT	RO	帧过滤缓冲区中的有效数据个数 单位为字节

在 bypass 模式下 (FSM_CTRL 0 的 fsm_cfg_en 和 fsm_work_en 同时为 0), 此计数器记录的是数据接收 FIFO 前所有 buf (环形缓冲区 + FSM_TABLE) 中缓冲的数据个数。在帧过滤模式下 (FSM_CTRL0 的 fsm_cfg_en 为 0 并且 fsm_work_en 为 1), 此计数器仅记录帧过滤环形缓冲区中的数据个数。

13.3.21 帧过滤比较字节表 (FSM_TABLE)

偏 移: 0x10 ~ 0x1f

复位值: 8' h00

表 13.25: 帧过滤比较字节表

位域	名称	访问	描述
7:0	fsm_table[i]	RW	帧过滤比较字节表 [i] fsm_table 0 – 15

fsm_table0 的地址偏移为 0x10 。其对应于帧前前导字符后的第 1 个字节。Fsm_table15 的地址偏移为 0x1f 。其对应于帧前前导字符后的第 16 个字节。Fsm_table 0 – 15 只有在 FSM_CTRL0 中的 fsm_cfg_en 位为 1 时才可以被软件访问。在 bypass 模式 (FSM_CTRL0 的 fsm_cfg_en 和 fsm_work_en 同时为 0) 下, FSM_TABLE 作为数据接收 FIFO 前的数据缓冲的一部分, 用于缓冲。

13.3.22 帧过滤广播地址表 (BR_TABLE)

偏 移： 0x20 ~ 0x27

复位值： 8' h00

表 13.26: 帧过滤广播地址表

位域	名称	访问	描述
7:0	br_table[i]	RW	帧过滤广播地址表 [i] br_table 0 - 7

br_table0 的地址偏移为 0x20。其对应于广播地址的第 1 个字节。 Br_table7 的地址偏移为 0x27。其对应于广播地址的第 8 个字节。

13.3.23 帧超时计数比较值的低 8 位 (FTIMEOUTL)

偏 移： 0x28

复位值： 8' h34

表 13.27: 帧超时计数比较值的低 8 位

位域	名称	访问	描述
7:0	Frame_timeout_cnt[7:0]	RW	帧超时计数比较值的低 8 位

13.3.24 帧超时计数比较值的高 8 位 (FTIMEOUTH)

偏 移： 0x29

复位值： 8' h01

表 13.28: 帧超时计数比较值的高 8 位

位域	名称	访问	描述
7:0	Frame_timeout_cnt[15:8]	RW	帧超时计数比较值的高 8 位

FTIMEOUTH 和 FTIMEOUTL 共同构成 Frame_timeout_cnt[15:0]。 Frame_timeout_cnt 用于配置在 32K 时钟下经历多少个时钟周期的空闲后，帧被认为结束。

13.3.25 帧过滤环形缓冲区的读指针 (BUF_RPTR)

偏 移： 0x2a

复位值： 8' h00

表 13.29: 帧过滤环形缓冲区的读指针

位域	名称	访问	描述
7:5	reserved	-	保留

位域	名称	访问	描述
4	listen_s	RW	过滤起始控制 当此位为 1 时，帧过滤需要从总线空闲开始
3:0	Buf_rptr	RO	帧过滤使用的环形缓冲区的读指针 该读指针在正常工作模式下（FSM_CTRL0 的 debug_mode、debug_mode_sync 位均为 0）随着数据从环形缓冲区进入接收数据 FIFO 而不断递增。在调试模式（FSM_CTRL0 的 debug_mode、debug_mode_sync 位均为 1）下，该读指针固定不动

环形缓冲区的读写指针是为调试准备的状态寄存器。其中，读指针在 FSM_CTRL0 的 debug_mode 、 debug_mode_sync 位同时为 1 时，才可以安全读取。当 listen_s 位为 1 时，在新的帧的前导符前，MBUS 总线必须处于总线空闲的状态（高电平）。这个总线空闲的最短时间由 FTIMEOUTL （偏移为 0x28）和 FTIMEOUTH （偏移为 0x29）的值决定。

13.3.26 帧过滤环形缓冲区数据直接读取空间（FSM_BUF）

偏 移： 0x30 ~ 0x3f

复位值： 8'h00

表 13.30: 帧过滤环形缓冲区数据直接读取空间

位域	名称	访问	描述
7:0	fsm_buf[i]	RO	帧过滤环形缓冲区 [i] fsm_buf 0 - 15

fsm_buf 0 的地址偏移为 0x30。 fsm_table 15 的地址偏移为 0x3f。 fsm_buf 0 - 15 只有在 FSM_CTRL 0 的 debug_mode 、 debug_mode_sync 位同时为 1 时才可以被软件访问。在读取此空间前可以通过查询 BUF_RPTR 获取环形缓冲区的读指针，通过读取 BUF_COUNT 获取当前环形缓冲区内部有效数据的个数。

13.4 配置流程

13.4.1 典型例子

假设 MBUS 总线收发时钟为 32768Hz （缺省配置）波特率为 2400，要对 CJ/T188-2004 的帧进行过滤。设备地址从 A0 到 A6 分别为：0x00、0x01、0x02、0x03、0x04、0x05、0x06。类型代码 T 为 0x01。需要执行下列初始化操作：

1. 打开分频器

- 置 LCR 的 dlab 位为 1
- 置 DL_D 的值为 0xb6
- 置 DL_H 的值为 0x0
- 置 DL_L 的值为 0x1

2. 配置 bit 窗口

- 置 sample_ctrl 的值为 0x38

3. 配置帧过滤参数

- 置 FSM_CTRL0 （地址偏移为 0x8）为 0x8
- 置 fsm_table0 （地址偏移为 0x10）为 0x68
- 置 fsm_table1 （地址偏移为 0x11）为 0x1
- 置 fsm_table2 （地址偏移为 0x12）为 0x0
- 置 fsm_table3 （地址偏移为 0x13）为 0x1
- 置 fsm_table4 （地址偏移为 0x14）为 0x2
- 置 fsm_table5 （地址偏移为 0x15）为 0x3
- 置 fsm_table6 （地址偏移为 0x16）为 0x4
- 置 fsm_table7 （地址偏移为 0x17）为 0x5
- 置 fsm_table8 （地址偏移为 0x18）为 0x6
- 置 PRE_CHAR （地址偏移为 0x9）为 0xfe

- 置 FSM_CTRL1 （地址偏移为 0xa）为 0x28
- 置 BR_MASK （地址偏移为 0xb）为 0x80
- 置 FSM_MASK （地址偏移为 0xc、0xd）为 0xfe00
- 置 BR_CTRL （地址偏移为 0xe）为 0x72
- 置 BR_TABLE （地址偏移为 0x20 ~ 0x27）为 0xaa
- 置 FSM_CTRL0 （地址偏移为 0x8）为 0x4

4. 配置 FCR

- 置 FCR 为 0x86 RX 接收的 trigger 值为 16，同时复位发送和接收数据通路

5. 开中断使能

- 置 IER 为 0x21 打开接收通路中断使能、打开帧结束中断使能

完成上述配置后。如果主机发送如下序列：fe fe 68 01 00 01 02 03 04 05 06 01 03 90 1f 00 39 16。MBUS 在看到中断后，可以从 DAT 端口读到的序列为：68 01 00 01 02 03 04 05 06 01 03 90 1f 00 39 16。如果主机发送如下序列（使用广播地址）：fe fe fe fe 68 01 aa aa aa aa aa aa aa 01 03 90 1f 00 39 16。MBUS 在看到中断后，可以从 DAT 端口读到的序列为：68 01 aa aa aa aa aa aa aa 01 03 90 1f 00 39 16。如果主机发送如下序列（使用广播地址，但是广播地址前需要匹配的字符错误）：fe fe fe fe 68 00 aa aa aa aa aa aa aa 01 03 90 1f 00 39 16。MBUS 控制器不会将上述序列的任何字符放入接收 FIFO 中，也不会产生中断。

13.4.2 特例

如果将典型例子中的 FSM_MASK （地址偏移为 0xc）置为 0x1 并且将 FSM_CTRL1 （地址偏移为 0xa）置为 0x0，此时任何接收到的数据都被认为匹配成功，MBUS 控制器将等同于将帧过滤功能关闭的 bypass 工作模式。在此配置下，接收到的数据经过环形缓冲区缓冲后直接被送入数据接收 FIFO。

此页留空

第十四章 实时时钟

14.1 概述

实时时钟模块提供年、月、日、时、分、秒、1/16 秒计数和一个定时中断。定时中断可将芯片从休眠状态唤醒。计时相关逻辑无复位电路，不会因为芯片复位而丢失时间信息。实时时钟的精度依赖于时钟源的精度，当时钟源的频率出现固定偏斜时，可通过修改分频系数来进行修正。

14.2 寄存器定义

实时时钟寄存器基址为 0x410。

表 14.1: 实时时钟寄存器列表

名称	偏移	描述
FREQ	0x00	分频值寄存器
CFG	0x04	配置寄存器
RTC0	0x08	时间值寄存器 0
RTC1	0x0c	时间值寄存器 1

14.2.1 分频值寄存器 (FREQ)

偏 移: 0x00

复位值: 无复位值

内部需要产生 1/16 秒的事件，通过本寄存器的配置分频得到 16Hz 时钟

表 14.2: 分频值寄存器

位域	名称	访问	描述
27:6	freqscale	RW	分频系数 27:16 位为整数部分，15:6 位为小数部分， $freqscale = \frac{freq_in}{16}$ ， freq_in 为输入时钟频率，以 Hz 为单位

14.2.2 配置寄存器 (CFG)

偏 移: 0x04

复位值: 32' h0

表 14.3: 配置寄存器

位域	名称	访问	描述
31	state	RW	操作进行状态 为 1 表示原子读写序列执行中，写 1 强制清零。仅供硬件调试使用
30	timer_en	RW	定时器使能 写 1 使能定时器，时间到后自动清此位
29:26	timer_month	RW	定时器月 表示对应单位的时间，用于定时器比较
25:21	timer_day	RW	定时器日 表示对应单位的时间，用于定时器比较
20:16	timer_hour	RW	定时器小时 表示对应单位的时间，用于定时器比较
15:10	timer_minute	RW	定时器分钟 表示对应单位的时间，用于定时器比较
9:4	timer_second	RW	定时器秒 表示对应单位的时间，用于定时器比较
3:0	timer_sixteenth	RW	定时器十六分之一秒 表示对应单位的时间，用于定时器比较

14.2.3 时间值寄存器 0 (RTC0)

偏 移: 0x08

复位值: 32' h0

表 14.4: 时间值寄存器 0

位域	名称	访问	描述
31	bad_time	RO	无效数值 表示所读出的时间无效
30:21	reserved	—	保留
20:16	hour	RW	小时 表示对应单位的时间，写时为待更新时间，读时为当前实时时钟时间
15:10	minute	RW	分钟 表示对应单位的时间，写时为待更新时间，读时为当前实时时钟时间
9:4	second	RW	秒 表示对应单位的时间，写时为待更新时间，读时为当前实时时钟时间
3:0	sixteenth	RW	十六分之一秒 表示对应单位的时间，写时为待更新时间，读时为当前实时时钟时间

14.2.4 时间值寄存器 1 (RTC1)

偏 移: 0x0c

复位值: 32' h0

读出此寄存器后必须接着读出 RTC0。

表 14.5: 时间值寄存器 1

位域	名称	访问	描述
31	bad_time	RO	无效数值 表示所读出的时间无效
30:16	reserved	—	保留
15:9	year	RW	年 表示对应单位的时间，写时为待更新时间，读时为当前实时时钟时间， 年计数自公元 2000 年起
8:5	month	RW	月 表示对应单位的时间，写时为待更新时间，读时为当前实时时钟时间
4:0	day	RW	日 表示对应单位的时间，写时为待更新时间，读时为当前实时时钟时间

14.3 说明

实时时钟的功能包括：

- 时间配置
配置时间时先后写 RTC0 和 RTC1 寄存器，中间不允许插入其它寄存器访问。
- 实时时间读取
直接读 RTC0 取得时分秒；或者先读 RTC1 后读 RTC0，中间不允许插入其它寄存器访问。内部采样寄存器周期性地从实时时钟中取得时间，当读出 bad_time 为 1 时说明采样未完成，应当再次读出。
- 定时中断
配置目标时间，然后使能定时器。定时器与实时时钟进行比较，相等时产生中断，使能信号自动清零。

此页留空

第十五章 定时器

15.1 概述

龙芯 1D100BA 提供了与 HPET 工作方式类似的 32 位定时器，支持单次定时和周期触发两种模式。

定时器包括 count、compare、step 三个寄存器，当时钟计数器 count 的计数值与 compare 相同时触发中断。在周期触发模式下，触发中断时 compare 自增 step 所存值。定时器工作在主时钟下，定时时间应根据主时钟频率进行设置。

HPET 的基址为 0x430，寄存器定义见下节。

15.2 寄存器空间

表 15.1: HPET 控制器寄存器列表

名称	偏移	描述
CFG	0x00	配置寄存器
CNT	0x04	计数值寄存器
CMP	0x08	比较值寄存器
STP	0x0c	步进值寄存器

15.2.1 配置寄存器 (CFG)

偏 移： 0x00

复位值： 32' h0

表 15.2: 配置寄存器

位域	名称	访问	描述
31:9	reserved	—	保留
8	int	RW	中断状态/清中断 读时表示中断状态，1 表示中断触发；写 1 时清中断状态
7:3	reserved	—	保留
2	periodic	RW	周期触发 置 1 使能周期触发，只在开始计数时有效
1	int_en	RW	中断使能 置 1 使能中断；不使能中断时中断状态和清除也有效，但处理器不会收到中断
0	start	RW	计数使能 置 1 开始计数，使能时无法修改 CNT/CMP/STP 寄存器的值

15.2.2 计数值寄存器 (CNT)

偏 移: 0x04

复位值: 32' h0

表 15.3: 计数值寄存器

位域	名称	访问	描述
31:0	count	RW	计数值 只能在未计数使能时修改; 触发中断时不会停止计数; 只在计数使能且非处理器调试模式下计数, 否则暂停

15.2.3 比较值寄存器 (CMP)

偏 移: 0x08

复位值: 32' hFFFFFFFF

表 15.4: 比较值寄存器

位域	名称	访问	描述
31:0	compare	RW	比较值 只能在未计数使能时修改; 周期中断触发时自增步进值

15.2.4 步进值寄存器 (STP)

偏 移: 0x0c

复位值: 32' h0

表 15.5: 步进值寄存器

位域	名称	访问	描述
31:0	step	RW	步进值 只能在未计数使能时修改

15.3 使用说明

例如, 在主时钟为 10MHz 时, 将定时器配置为首次计时 2 秒、步进 5 秒并打开中断 (不包含处理器和中断控制器相关寄存器的配置)。其寄存器配置和中断处理句柄如下所示。

配置 HPET 寄存器

```
CFG = 0; // 清零配置寄存器
/* 使能 Confreg 中相关中断位 */
CNT = 0; // 清零计数值
CMP = 20000000; // 比较值设为 2 秒
STP = 50000000; // 步进值设为 5 秒
CFG = 0x7; // 开始计数, 中断使能, 周期触发
```

```
                                中断处理句柄
CFG = CFG;                      // 读出中断状态并写回，用于清除中断
/* 清除 Confreg 中相关中断位 */
/* 进行定时相关工作，如喂看门狗 */
return 0;
```

修订历史

序号	版本号	更新内容
1	V1.0	1D100BA 发布版

技术支持

可通过邮箱向我司提交芯片手册和产品使用的问题，并获取技术支持。

服务邮箱：service@loongson.cn

声明

本文档版权归龙芯中科技术股份有限公司所有，并保留一切权利。未经许可不得擅自实施传播等侵害版权人合法权益的行为。本文档仅提供阶段性信息，所含内容可根据产品的实际情况随时更新，恕不另行通知。如因文档使用不当造成的直接或间接损失，本公司不承担任何责任。

龙芯中科技术股份有限公司

Loongson Technology Corporation Limited

地址：北京市海淀区中关村环保科技示范园龙芯产业园 2 号楼

Building No.2, Loongson Industrial Park, Zhongguancun Environmental Protection Park

电话 (Tel) : 010-62546668

传真 (Fax) : 010-62600826