

四串口芯片 CH9434

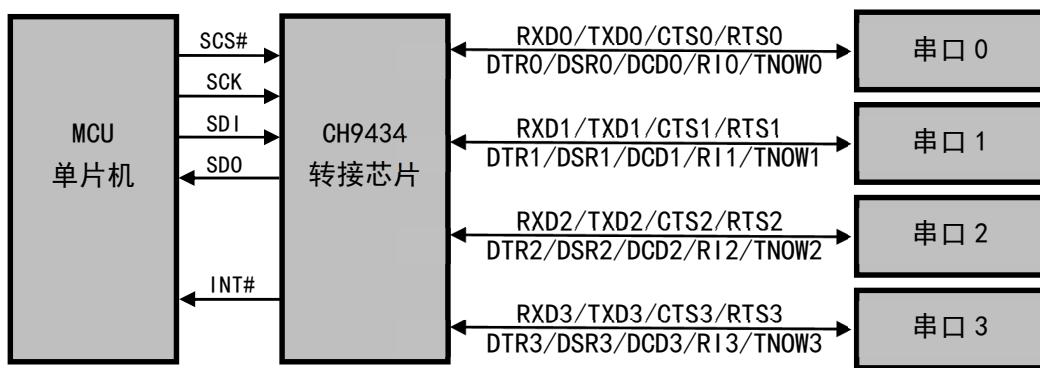
手册

版本: 1.3

<https://wch.cn>

1、概述

CH9434 是一款 SPI 转四串口转接芯片，提供四组全双工的 9 线异步串口，用于单片机/嵌入式系统扩展异步串口。CH9434 包含四个兼容 16C550 的异步串口，最高支持 4Mbps 波特率通讯。最多支持 25 路 GPIO，提供半双工收发自动切换引脚 TNOW。



2、特点

- 工作电压: 3.3V
- 支持通讯波特率设置，波特率范围 1200–4000000bps
- 串口每个方向独立 FIFO 缓存，发送 1536 字节，接收 2048 字节
- 完全独立四个异步串口，兼容 16C550 并且有所增强
- 串口支持 5/6/7/8 个数据位以及 1/2 个停止位
- 串口支持奇、偶、无校验、空白 0、标志 1 等校验方式
- 支持常用的 MODEM 联络信号 RTS、DTR、DCD、RI、DSR、CTS
- 提供半双工 RS485 收发使能引脚
- SPI 最高可达为 32Mbit/s
- 支持低功耗睡眠模式，可通过 SPI 接口唤醒
- 芯片提供可配置 GPIO 功能
- 外接 32M 晶体提供时钟
- QFN48 无铅封装，兼容 RoHS

3、应用领域

- MCU/DSP/嵌入式系统
- 工业自动化 RS-485 通讯
- 串口服务器、多串口卡
- 与蓝牙、4G、Wi-Fi 等串口模块通讯实现无线传输

4、引脚排列

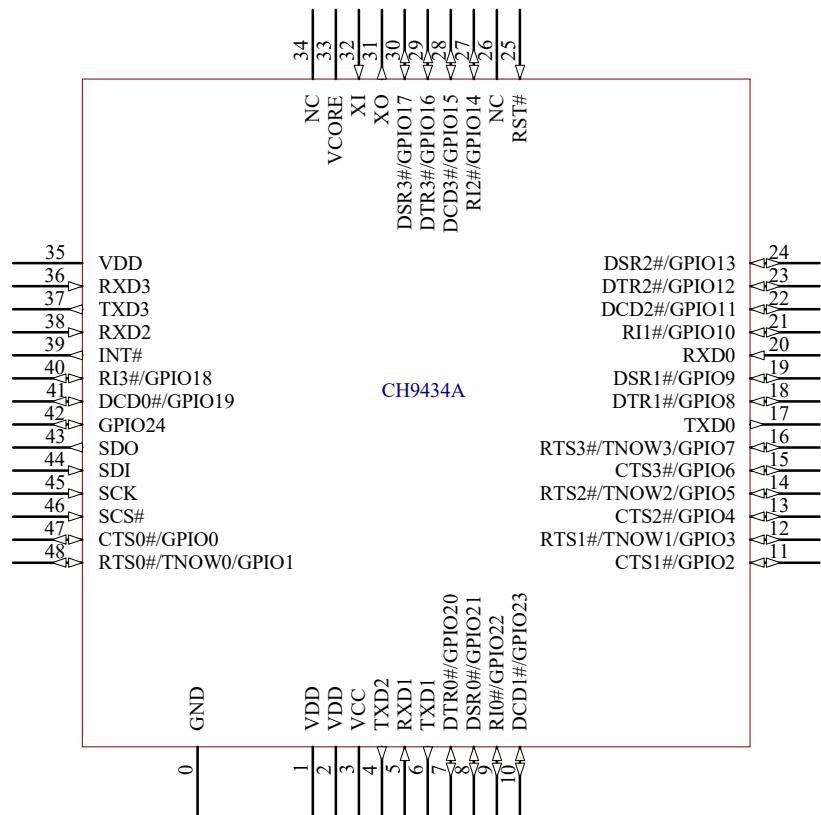


表 4-1 封装说明

封装形式	塑体尺寸	引脚节距		封装说明	订货型号
QFN48	5*5mm	0. 35mm	13. 8mil	四边无引线 48 脚	CH9434A

注: CH9434A 基于 CH9434M 升级, 引脚相同, 封装兼容。

5、引脚定义

表 5-1 CH9434A 引脚定义

引脚号	引脚名称	引脚类型	引脚说明
0	GND	P	电源地。
1	V _{DD}	P	芯片内部电源, 与其他 V _{DD} 短接, 需外接退耦电容。 电容值建议不小于 0.1uF。
2	V _{DD}	P	芯片内部电源, 与其他 V _{DD} 短接。
3	V _{CC}	P	芯片电源输入, 需要外接不小于 0.1uF 电源退耦电容。
4	TXD2	0	UART2 的串行数据输出。
5	RXD1	I	UART1 的串行数据输入。
6	TXD1	0	UART1 的串行数据输出。
7	DTR0#/GPIO20	I/O	DTR0#: UART0 的 MODEM 输出信号, 数据终端就绪。 GPIO20: 通用双向数字 I/O 引脚。
8	DSR0#/GPIO21	I/O	DSR0#: UART0 的 MODEM 输入信号, 数据装置就绪。 GPIO21: 通用双向数字 I/O 引脚。
9	RIO#	I/O	RIO#: UART0 的 MODEM 输入信号, 振铃指示。

	/GPIO22		GPIO22: 通用双向数字 I/O 引脚。
10	DCD1# /GPIO23	I/O	DCD1#: UART1 的 MODEM 输入信号, 载波检测。 GPIO23: 通用双向数字 I/O 引脚。
11	CTS1# /GPIO2	I/O	CTS1#: UART1 的 MODEM 联络输入信号, 清除发送, 低电平有效。 GPIO2: 通用双向数字 I/O 引脚。
12	RTS1# /TNOW1 /GPIO3	I/O	RTS1#: UART1 的 MODEM 联络输出信号, 请求发送, 低电平有效。 TNOW1: UART1 的 RS485 收发切换控制引脚。 GPIO3: 通用双向数字 I/O 引脚。
13	CTS2# /GPIO4	I/O	CTS2#: UART2 的 MODEM 联络输入信号, 清除发送, 低电平有效。 GPIO4: 通用双向数字 I/O 引脚。
14	RTS2# /TNOW2 /GPIO5	I/O	RTS2#: UART2 的 MODEM 联络输出信号, 请求发送, 低电平有效。 TNOW2: UART2 的 RS485 收发切换控制引脚。 GPIO5: 通用双向数字 I/O 引脚。
15	CTS3# /GPIO6	I/O	CTS3#: UART3 的 MODEM 联络输入信号, 清除发送, 低电平有效。 GPIO6: 通用双向数字 I/O 引脚。
16	RTS3# /TNOW3 /GPIO7	I/O	RTS3#: UART3 的 MODEM 联络输出信号, 请求发送, 低电平有效。 TNOW3: UART3 的 RS485 收发切换控制引脚。 GPIO7: 通用双向数字 I/O 引脚。
17	TXD0	0	UART0 的串行数据输出。
18	DTR1 /GPIO8	I/O	DTR1#: UART1 的 MODEM 输出信号, 数据终端就绪。 GPIO8: 通用双向数字 I/O 引脚。
19	DSR1 /GPIO9	I/O	DSR1#: UART1 的 MODEM 输入信号, 数据装置就绪。 GPIO9: 通用双向数字 I/O 引脚。
20	RXD0	I	UART0 的串行数据输入。
21	RI1# /GPIO10	I/O	RI1#: UART1 的 MODEM 输入信号, 振铃指示。 GPIO10: 通用双向数字 I/O 引脚。
22	DCD2# /GPIO11	I/O	DCD2#: UART2 的 MODEM 输入信号, 载波检测。 GPIO11: 通用双向数字 I/O 引脚。
23	DTR2# /GPIO12	I/O	DTR2#: UART2 的 MODEM 输出信号, 数据终端就绪。 GPIO12: 通用双向数字 I/O 引脚。
24	DSR2# /GPIO13	I/O	DSR2#: UART2 的 MODEM 输入信号, 数据装置就绪。 GPIO13: 通用双向数字 I/O 引脚。
25	RST	I	芯片复位引脚, 低电平有效。
26	NC	N	无效引脚, 悬空。
27	RI2# /GPIO14	I/O	RI2#: UART2 的 MODEM 输入信号, 振铃指示。 GPIO14: 通用双向数字 I/O 引脚。
28	DCD3# /GPIO15	I/O	DCD3#: UART3 的 MODEM 输入信号, 载波检测。 GPIO15: 通用双向数字 I/O 引脚。
29	DTR3# /GPIO16	I/O	DTR3#: UART3 的 MODEM 输出信号, 数据终端就绪。 GPIO16: 通用双向数字 I/O 引脚。
30	DSR3# /GPIO17	I/O	DSR3#: UART3 的 MODEM 输入信号, 数据装置就绪。 GPIO17: 通用双向数字 I/O 引脚。
31	XO	A	高频振荡器的反相输出端。
32	XI	A	高频振荡器的反相输入端。

33	V_{CORE}	P	芯片内部的电源，需贴近引脚外接退耦电容。 建议不小于 0.1uF。
34	NC	N	无效引脚，悬空。
35	V_{DD}	P	芯片内部电源，与其他 V_{DD} 短接，需外接退耦电容。 电容值建议不小于 0.1uF。
36	RXD3	I	UART3 的串行数据输入。
37	TXD3	O	UART3 的串行数据输出。
38	RXD2	I	UART2 的串行数据输入。
39	INT#	O	中断输出引脚，低电平有效。
40	R13# /GPIO18	I/O	R13#: UART3 的 MODEM 输入信号，振铃指示。 GPIO18: 通用双向数字 I/O 引脚。
41	DCD0# /GPIO19	I/O	DCD0#: UART0 的 MODEM 输入信号，载波检测。 GPIO19: 通用双向数字 I/O 引脚。
42	GPIO24	I/O	GPIO24: 通用双向数字 I/O 引脚。
43	SDO	O	SPI 串行数据输出。
44	SDI	I	SPI 串行数据输入。
45	SCK	I	SPI 串行时钟输入。
46	SCS#	I	SPI 片选输入，低电平有效。
47	CTS0# /GPIO0	I/O	CTS0#: UART0 的 MODEM 联络输入信号，清除发送，低电平有效。 GPIO0: 通用双向数字 I/O 引脚。
48	RTS0# /TNOW0 /GPIO1	I/O	RTS0#: UART0 的 MODEM 联络输出信号，请求发送，低电平有效。 TNOW0: UART0 的 RS485 收发切换控制引脚。 GPIO1: 通用双向数字 I/O 引脚。

注: P = 电源引脚; I = 输入引脚; O = 输出引脚; N = 空脚; 复用功能在前优先。

6、时钟配置

CH9434 需外接一个晶体配合芯片内部的时钟振荡器提供 32MHz 输入时钟。串口基准时钟可由输入时钟直接产生或开启时钟倍频再分频产生，提供给串口 0~串口 3，倍频系数固定为 15，分频系数通过 R8_CLK_CTRL_CFG 寄存器进行配置。

开启时钟倍频时串口基准时钟 = 32MHz * 倍频系数/分频系数，使用常规波特率时建议分频系数配置为 13。初始化 CH9434 时可以对芯片时钟方式进行配置，配置后需延时一段时间后进行其他操作。

7、寄存器

7.1 串口寄存器

CH9434 芯片提供 4 个独立的串口模块，分别使用独立的 8 个字节寄存器进行配置。其地址分别为：00H-07H 为串口 0、10H-17H 为串口 1、20H-27H 为串口 2、30H-37H 为串口 3。串口寄存器兼容工业标准 16C550 或者 16C750 并有所增强。表中 DLAB 表示寄存器 LCR 中的位 7，X 表示不关心 DLAB 值，R0 表示寄存器只读，W0 表示寄存器只写，R/W 表示寄存器可读可写。

表 7-1 串口寄存器

地址	DLAB	R/W	名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0	0	R0	RBR	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0	0	W0	THR	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
1	0	R/W	IER	RESET	LOWPOWER	0	0	IEMODEM	IELINES	IETHRE	IERECV
2	X	R0	IIR	FIFOENS	FIFOENS	0	0	IID3	IID2	IID1	NOINT
2	X	W0	FCR	RECVTG1	RECVTG0	0	0	0	TFIFORST	RFIFORST	FIFOEN
3	X	R/W	LCR	DLAB	BREAKEN	PARMODE1	PARMODE0	PAREN	STOPBIT	WORDSZ1	WORDSZ0
4	X	R/W	MCR	0	0	AFE	LOOP	OUT2	OUT1	RTS	DTR
5	X	R0	LSR	RFIFOERR	TEMPT	THRE	BREAKINT	FRAMEERR	PARERR	OVERR	DATARDY
6	X	R0	MSR	DCD	RI	DSR	CTS	△DCD	△RI	△DSR	△CTS
7	X	R/W	SCR	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
0	1	R/W	DLL	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
1	1	R/W	DLM	位 15	位 14	位 13	位 12	位 11	位 10	位 9	位 8

下表是串口寄存器在上电复位或者串口软复位之后的默认值。

表 7-2 默认值

寄存器名称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
IER	0	0	0	0	0	0	0	0
IIR	0	0	0	0	0	0	0	1
FCR	0	0	0	0	0	0	0	0
LCR	0	0	0	0	0	0	0	0
MCR	0	0	0	0	0	0	0	0
LSR	0	1	1	0	0	0	0	0
MSR	DCD	RI	DSR	CTS	0	0	0	0
SCR	保持	保持	保持	保持	保持	保持	保持	保持
FIFO	复位，包括发送 FIFO 和接收 FIFO							
TSR	复位，TSR 是串口发送移位寄存器							
RSR	复位，RSR 是串口接收移位寄存器							

其它	未定义
----	-----

RBR: 接收缓冲寄存器，如果 LSR 的 DATARDY 位为 1 则可以从该寄存器读取接收到的数据。如果 FIFOEN 为 1 则从串口移位寄存器 RSR 接收到的数据首先被存放于接收 FIFO 中，然后通过该寄存器读出。

THR: 发送保持寄存器，包括发送 FIFO，用于写入准备发送的数据。如果 FIFOEN 为 1 则写入的数据首先被存放于发送 FIFO 中，然后通过发送移位寄存器 TSR 逐个输出。

IER: 中断使能寄存器，包括增强功能控制位以及串口中断使能。

RESET: 该位置 1 则软复位该串口，该位能够自动清 0，无需软件清 0。

LOWPOWER: 该位为 1 则关闭该串口的内部基准时钟，从而使该串口进入低功耗状态。

IEMODEM: 该位为 1 则允许调制解调器输入状态变化中断。

IELINES: 该位为 1 则允许接收线路状态中断。

IETHRE: 该位为 1 则允许发送保持寄存器空中断。

IERECV: 该位为 1 则允许接收到数据中断。

IIR: 中断识别寄存器，用于分析中断源并处理。

FIFOENS: 该位为 FIFO 启用状态，为 1 表示已经启用 FIFO。

表 7-3 IIR 寄存器位

IIR 寄存器位				优先级	中断类型	中断源	清中断方法
IID3	IID2	IID1	NOINT				
0	0	0	1	无	没有中断产生	没有中断	
0	1	1	0	1	接收线路状态	OVERR、PARERR、FRAMEERR、BREAKINT	读 LSR
0	1	0	0	2	接收数据可用	接收到的字节数达到 FIFO 的触发点	读 RBR
1	1	0	0	2	接收数据超时	超过 4 个数据的时间未收到下一数据	读 RBR
0	0	1	0	3	THR 寄存器空	发送保持寄存器空，IETHRE 从 0 变 1 可以重新使能中断	读 IIR 或写 THR
0	0	0	0	4	MODEM 输入变化	△CTS、△DSR、△RI、△DCD	读 MSR

FCR: 先进先出缓冲区 FIFO 控制寄存器，用于使能和复位 FIFO。

RECVTG1 和 **RECVTG0:** 设置接收 FIFO 的中断和硬件流控制的触发点，00 对应 256 个字节，即接收满 256 个字节产生接收数据可用的中断，并在使能硬件流控制时自动无效 RTS 引脚，01 对应 512 个字节，10 对应 1024 个字节，11 对应 1280 个字节。

TFIFORST: 该位置 1 则清空发送 FIFO 中的数据（不含 TSR），该位能够自动清 0，无需软件清 0。

RFIFORST: 该位置 1 则清空接收 FIFO 中的数据（不含 RSR），该位能够自动清 0，无需软件清 0。

FIFOEN: 该位为 1 则启用 FIFO，该位清 0 则禁用 FIFO，禁用 FIFO 后为 160450 兼容模式，相当于 FIFO 只有一个字节。

LCR: 线路控制寄存器，用于控制串口通讯的格式。

DLAB: 该位为除数锁存器存取使能，为 1 时才能存取 DLL 和 DLM，为 0 时才能存取 RBR/THR/IER。

BREAKEN: 该位为 1 则强制产生 BREAK 线路间隔。

PARMODE1 和 **PARMODE0:** 当 PAREN 为 1 时设置奇偶校验位的格式：00 则奇校验，01 则偶校验，10 则标志位（MARK，置 1），11 则空白位（SPACE，清 0）。

PAREN: 该位为 1 则允许发送时产生和接收时校验奇偶校验位，为 0 则无奇偶校验位。

STOPBIT: 该位为 1 则两个停止位，为 0 则一个停止位。

WORDSZ1 和 **WORDSZ0:** 设置字长度，00 则 5 个数据位，01 则 6 个数据位，10 则 7 个数据位，

11 则 8 个数据位。

MCR: 调制解调器 MODEM 控制寄存器，用于控制 MODEM 输出。

AFE: 该位为 1 则允许 CTS 和 RTS 硬件自动流控制。如果 AFE 为 1，那么仅在检测到 CTS 引脚输入有效（低电平有效）时串口才继续发送下一个数据，否则暂停串口发送。如果 AFE 为 1 并且 RTS 为 1，那么当接收 FIFO 空时，串口会自动有效 RTS 引脚（低电平有效），直到接收的字节数达到 FIFO 的触发点时，串口才自动无效 RTS 引脚，并能够在接收 FIFO 空时再次有效 RTS 引脚。使用硬件自动控制，可将己方的 CTS 引脚接到对方的 RTS 引脚，并将己方的 RTS 引脚送到对方的 CTS 引脚。

LOOP: 该位为 1 则使能内部回路的测试模式。在内部回路的测试模式下，串口所有对外输出引脚均为无效状态，TXD 内部返回到 RXD（即 TSR 的输出内部返回到 RSR 的输入），RTS 内部返回到 CTS，DTR 内部返回到 DSR，OUT1 内部返回到 RI，OUT2 内部返回到 DCD。

OUT2: 该位为 1 则允许该串口的中断请求输出，否则该串口不产生实际中断请求。

OUT1: 该位为用户可定义 MODEM 控制位，没有连接实际输出引脚。

RTS: 该位为 1 则 RTS 引脚输出有效（低电平有效），否则 RTS 引脚输出无效。

DTR: 该位为 1 则 DTR 引脚输出有效（低电平有效），否则 DTR 引脚输出无效。

LSR: 线路状态寄存器，用于查询方式分析串口状态。

RFIFOERR: 该位为 1 表示在接收 FIFO 中存在至少一个 PARERR、FRAMEERR 或 BREAKINT 错误。

TEMT: 该位为 1 表示发送保持寄存器 THR 和发送移位寄存器 TSR 全空。

THRE: 该位为 1 表示发送保持寄存器 THR 空。

BREAKINT: 该位为 1 表示检测到 BREAK 线路间隔。

FRAMEERR: 该位为 1 表示正在从接收 FIFO 中读取的数据的帧错误，缺少有效的停止位。

PARERR: 该位为 1 表示正在从接收 FIFO 中读取的数据的奇偶校验错。

OVERR: 该位为 1 表示接收 FIFO 缓冲区溢出。

DATARDY: 该位为 1 表示接收 FIFO 中有接收到的数据，读取 FIFO 中所有数据后，该位自动清 0。

MSR: 调制解调器 MODEM 状态寄存器，用于查询 MODEM 状态。

DCD: 该位是 DCD 引脚的位反，为 1 表示 DCD 引脚有效（低电平有效）。

RI: 该位是 RI 引脚的位反，为 1 表示 RI 引脚有效（低电平有效）。

DSR: 该位是 DSR 引脚的位反，为 1 表示 DSR 引脚有效（低电平有效）。

CTS: 该位是 CTS 引脚的位反，为 1 表示 CTS 引脚有效（低电平有效）。

△DCD: 该位为 1 表示 DCD 引脚输入状态发生过变化。

△RI: 该位为 1 表示 RI 引脚输入状态发生过变化。

△DSR: 该位为 1 表示 DSR 引脚输入状态发生过变化。

△CTS: 该位为 1 表示 CTS 引脚输入状态发生过变化。

SCR: 用户可定义寄存器。

DLL 和 DLM: 波特率除数锁存器，DLL 是低字节，DLM 是高字节，两者组成的 16 位除数用于由 16 位计数器构成的串口波特率产生器。该除数=串口基准时钟/8/所需通讯波特率。

7.2 接口寄存器

表 7-4 接口寄存器

地址	名称	功能	位	访问	位定义	默认值
41H	R8_TNOW_CTRL_CFG	TNOW 功能设置	[7:4]	R/W	TNOW 引脚电平反相	0000b
			[3:0]	R/W	TNOW 引脚功能控制	0000b
42H	R8_FIFO_CTRL	FIFO 计数器设置	[7:5]	R0	保留	000b
			4	R/W	收发缓存区控制	0

					0: 接收 FIFO; 1: 发送 FIFO。	
			[3:0]	R/W	串口号	0000b
43H	R8_FIFO_CNT_L	FIFO 计数器	[7:0]	R0	FIFO 计数器低 8 位	XXh
44H	R8_FIFO_CNT_H	FIFO 计数器	[7:0]	R0	FIFO 计数器高 8 位	XXh

R8_TNOW_CTRL_CFG: 设置 TNOW 功能开启和极性，极性默认发送数据时，对应 TNOW 引脚拉高，发送结束 TNOW 拉低。

R8_FIFO_CTRL: FIFO 计数器设置，设置 FIFO 的方向以及串口号后，读取 R8_FIFO_CNT_L 和 R8_FIFO_CNT_H 的值后对应读取的 FIFO 计数器。其中：读取串口发送 FIFO 计数器值为当前剩余 FIFO 数量，主控可以根据这个值填入最大数量的发送数据；读取接收 FIFO 计数器值位当前已接收数据数量，主控可以根据这个值一次将串口数据全部读走。

7.3 时钟电源寄存器

表 7-5 时钟电源寄存器

地址	名称	功能	位	访问	位定义	默认值
48H	R8_CLK_CTRL_CFG	芯片时钟设置	[7:6]	R/W	HCLK 系统时钟源模式选择： 0x: 外部晶振 1x: 使用外部晶振并开启倍频	00b
			5	R0	保留	0
			[4:0]	R/W	分频系数	00000b
4AH	R8_SLEEP_MOD_CFG	睡眠功能设置	[7:3]	R0	保留	00000b
			[2:0]	R/W	芯片低功耗设置 0: 不进入低功耗状态； 1: SLEEP 状态； 2-7: 保留。	000b

R8_CLK_CTRL_CFG: 芯片时钟设置，默认使用外部晶振提供的 32MHz 时钟。分频系数是在开启倍频功能后启用。

7.4 GPIO 寄存器

表 7-6 GPIO 寄存器

地址	名称	功能	位	访问	位定义	默认值
50H	R8_GPIO_FUNC_EN_0	GPIO7-0 使能	[7:0]	R/W	[7:0] 对应 GPIO7-0 使能控制 0: 关闭 GPIO 功能 1: 开启 GPIO 功能	00h
51H	R8_GPIO_FUNC_EN_1	GPIO15-8 使能	[7:0]	R/W	[7:0] 对应 GPIO15-8 使能控制，值定义同上	00h
52H	R8_GPIO_FUNC_EN_2	GPIO23-16 使能	[7:0]	R/W	[7:0] 对应 GPIO23-16 使能控制，值定义同上	00h
53H	R8_GPIO_FUNC_EN_3	保留	[7:1]	R0	保留	00000000b
		GPIO24 使能	0	R/W	对应 GPIO24 使能控制，值定义同上	0b
54H	R8_GPIO_DIR_MOD_0	GPIO7-0 方向设置	[7:0]	R/W	[7:0] 对应 GPIO7-0 方向设置 0: GPIO 设置成输入	00h

					1:GPIO 设置成输出	
55H	R8_GPIO_DIR_MOD_1	GPIO15-8 方向设置	[7:0]	R/W	[7:0] 对应 GPIO15-8 方向设置, 值定义同上	00h
56H	R8_GPIO_DIR_MOD_2	GPIO23-16 方向设置	[7:0]	R/W	[7:0] 对应 GPIO23-16 方向设置, 值定义同上	00h
57H	R8_GPIO_DIR_MOD_3	保留	[7:1]	RO	保留	00000000b
		GPIO24 方向设置	0	R/W	对应 GPIO24 方向设置, 值定义同上	0b
58H	R8_GPIO_PU_MOD_0	GPIO7-0 上拉电阻设置	[7:0]	R/W	[7:0] 对应 GPIO7-0 上拉电阻设置 0:关闭 GPIO 上拉电阻 1:开启 GPIO 上拉电阻	00h
59H	R8_GPIO_PU_MOD_1	GPIO15-8 上拉电阻设置	[7:0]	R/W	[7:0] 对应 GPIO15-8 上拉电阻设置, 值定义同上	00h
5AH	R8_GPIO_PU_MOD_2	GPIO23-16 上拉电阻设置	[7:0]	R/W	[7:0] 对应 GPIO23-16 上拉电阻设置, 值定义同上	00h
5BH	R8_GPIO_PU_MOD_3	保留	[7:1]	RO	保留	00000000b
		GPIO24 上拉电阻设置	0	R/W	对应 GPIO24 上拉电阻设置, 值定义同上	0b
5CH	R8_GPIO_PD_MOD_0	GPIO7-0 下拉电阻设置	[7:0]	R/W	[7:0] 对应 GPIO7-0 下拉电阻设置 0:关闭 GPIO 下拉电阻 1:开启 GPIO 下拉电阻	00h
5DH	R8_GPIO_PD_MOD_1	GPIO15-8 下拉电阻设置	[7:0]	R/W	[7:0] 对应 GPIO15-8 下拉电阻设置, 值定义同上	00h
5EH	R8_GPIO_PD_MOD_2	GPIO23-16 下拉电阻设置	[7:0]	R/W	[7:0] 对应 GPIO23-16 下拉电阻设置, 值定义同上	00h
5FH	R8_GPIO_PD_MOD_3	保留	[7:1]	RO	保留	00000000b
		GPIO24 下拉电阻设置	0	R/W	对应 GPIO24 下拉电阻设置, 值定义同上	0b
60H	R8_GPIO_PIN_VAL_0	GPIO7-0 输入输出电平	[7:0]	R/W	[7:0] 对应 GPIO7-0 引脚电平 写入寄存器 GPIO 为输出时: 1: 输出高电平 0: 输出低电平 读取寄存器 GPIO 为输入时: 该引脚电平状态	00h
61H	R8_GPIO_PIN_VAL_1	GPIO15-8 输入输出电平	[7:0]	R/W	[7:0] 对应 GPIO15-8 引脚电平, 值定义同上	00h
62H	R8_GPIO_PIN_VAL_2	GPIO23-16 输入输出电平	[7:0]	R/W	[7:0] 对应 GPIO23-16 引脚电平, 值定义同上	00h
63H	R8_GPIO_PIN_VAL_3	保留	[7:1]	RO	保留	00000000b
		GPIO24 输入输出电平	0	R/W	对应 GPIO24 引脚电平, 值定义同上	0b

GPIO 设置方式:

(1) 打开对应 GPIO 使能位;

(2) 设置上下拉电阻配置以及方向；

(3) 设置或者读取“输入输出电平”寄存器，当设置该寄存器，GPIO 方向为输出时处理对应位，并输出对应的电平值，输入引脚不关心对应位；当读取该寄存器，GPIO 方向为输入时处理对应位，输出引脚不关心对应位。

芯片不含的 GPIO，则相关寄存器定义无效。

7.5 扩展寄存器

表 7-7 扩展寄存器

地址	名称	功能	位	访问	位定义	默认值
64H	EXT_SPI_CONT_MODE	SPI 传输模式	[7:1]	R/W	保留	0
			0	R/W	RBR 支持连续模式 0：关闭 RBR 连续读取模式 1：开启 RBR 连续读取模式	0
65H	CHIP_VER_ADD	当前固件版本	[7:0]	R0	VER[0]：当前版本号低 8 位	XXh
			[7:0]	R0	VER[1]：当前版本号高 8 位	XXh
			[7:0]	R0	VER[2]：版本信息校验	XXh
			[7:0]	R0	VER[3]：结束字符	5Ah

EXT_SPI_CONT_MODE 和 CHIP_VER_ADD 寄存器是版本 v1.02 开始支持；固件版本定义详参 8.8 章节。

8、功能说明

8.1 中断与查询

CH9434 芯片的 INT#引脚为中断请求输出引脚，默认引脚在芯片上电后为上拉输出模式，低电平有效。由于四路串口共用一个中断引脚输出，所以当有中断有效信号时，主控 MCU 需要查询所有串口的中断状态以分析是哪个串口有中断请求。

如果串口有多个中断请求输出，当主控 MCU 读取一个有效的中断状态后，CH9434 会将中断引脚暂时拉高，之后再拉低。主控 MCU 可以查询判断，只要中断引脚为低电平就直接进行串口中断状态查询。

8.2 串口操作

串口的操作提供了 IIR 和 LSR 寄存器查询串口的收发状态，CH9434 还扩展了一个 FIFO 寄存器用于直接获取当前芯片的串口收发数据量。

当串口中断查询到有接收数据后，可以先读取当前串口的接收 FIFO 长度，然后直接操作 RBR 寄存器读取所有数据。主控 MCU 也可以简化处理直接定时查询 FIFO 当前接收数据数量，然后根据接收 FIFO 的数量操作 RBR 寄存器读取数据。

发送数据可以查询 IIR 和 LSR 寄存器的发送缓存区空状态，然后操作 THR 寄存器进行串口发送数据。主控 MCU 也可以简化处理使用查询 FIFO 大小进行数据传输，查询发送 FIFO 数据长度的值为剩余 FIFO 大小，主控 MCU 可以此为大小进行将发送数据依次填入 THR 寄存器进行数据发送。

串口流控功能使能是将 AFE 位置 1，CH9434 将自动进行硬件流控。芯片将自动根据 FIFO 大小对流控引脚进行操作。启用自动流控后，CTS 有效时芯片串口将连续发送数据，CTS 引脚无效时，串口最多发送 8 字节数据后停止发送。RTS 在触发 FIFO 达到设定的流控字节数目后自动失效。

8.3 RS485切换引脚TNOW

CH9434 串口提供 RS485 切换引脚 TNOW，引脚功能与其他功能复用，当启用 TNOW 功能后，将自动失效原 MODEM 信号功能。TNOW 还支持极性调节，以适应不同的极性使用场景。

8.4 GPIO 功能

CH9434 支持功能引脚复用为 GPIO 功能，最多支持 25 路，每个 IO 都可以独立设置方向、上拉电阻和下拉电阻配置。启用 GPIO 功能后，将自动失效该 IO 其他复用功能。GPIO 功能在设置时需要注意该 IO 原来功能的模式，设置时需要注意 DIR、PD、PU 等 IO 寄存器的设置顺序防止 IO 出现“抖动”。

8.5 低功耗模式

CH9434 支持低功耗模式设置。设置 SLEEP 模式后，芯片暂停运行，电流功耗可降至 4.8mA，可以直接操作 SPI 进行唤醒。

8.6 SPI 串行接口

SPI 同步串行接口信号线包括：SPI 片选输入引脚 SCS#、串行时钟输入引脚 SCK、串行数据输入引脚 SDI 和串行数据输出引脚 SDO。SPI 数据位序：高位在前。发送数据格式为第一个字节为操作地址，第二个字节为写入的数据或者读取的数据，操作地址的最高位为操作位，操作位为 1 则为写入数据，为 0 则为读取数据。当写入数据时，地址和数据两个字节间需要 1uS 延时，发送完数据后需要延时 3uS 才可以进行下一次操作。当读取数据时，地址和数据需要延时 3uS，即发送完地址后延时 3uS 进行数据读取。

8.7 扩展寄存器

命令在版本 v1.02 版本后支持。

开启 RBR 连续读取模式功能后，SPI 接口频率需设置成 2MHz，在数据接收中断触发后，主控 MCU

读取当前 FIFO 接收长度 len ，读取 RBR 时，按照 $\text{len}+1$ 长度进行连续读取，读取完成后删除第一个字节即可。此模式推荐在系统应用无法控制 μs 延时下使用，可以控制延时的系统或主控 MCU 可以直接通过延时去连续读取。具体的功能可参考相关例子程序。

RBR 接口读取流程 ($\text{len}=n$)：RBR+DATA1+DATA2+…+DATAn+DATAn+1，地址和数据之间的延时和数据之间的延时需要去掉，流程结束后保存 DATA2 到 DATAn+1 数据。

8.8 版本号

版本号寄存器在 v1.02 版本开始增加，用于区分新增功能。版本号读取需要读取 4 字节版本信息，其中 VER[0] 和 VER[1] 为版本信息，解析格式为 BCD 码，VER[1] 字段高 4 位为保留信息（固定为 0），VER[2] 为版本信息校验字段，值为 (VER[0]+VER[1])，VER[3] 为结束字段，值固定为 0x5A。

版本号读取后，主控 MCU 需对版本号与 VER[2] 进行校验以确认版本号的有效性。若校验字段解析失败，则芯片版本号视为低于 v1.02；若校验成功，则表明芯片包含有效的版本信息。

9、参数

9.1 绝对最大值（临界或者超过绝对最大值将可能导致芯片工作不正常甚至损坏）

名称	参数说明	最小值	最大值	单位
T _A	工作时的环境温度	-40	85	°C
T _S	存储时的环境温度	-40	105	°C
V _{CC}	电源电压 (V _{CC} 接电源, GND 接地)	-0.4	3.9	V
V _{IO}	输入或者输出引脚上的电压	-0.4	V _{CC} +0.4	V

9.2 电气参数（测试条件：T_A = 25°C, V_{CC} = 3.3V）

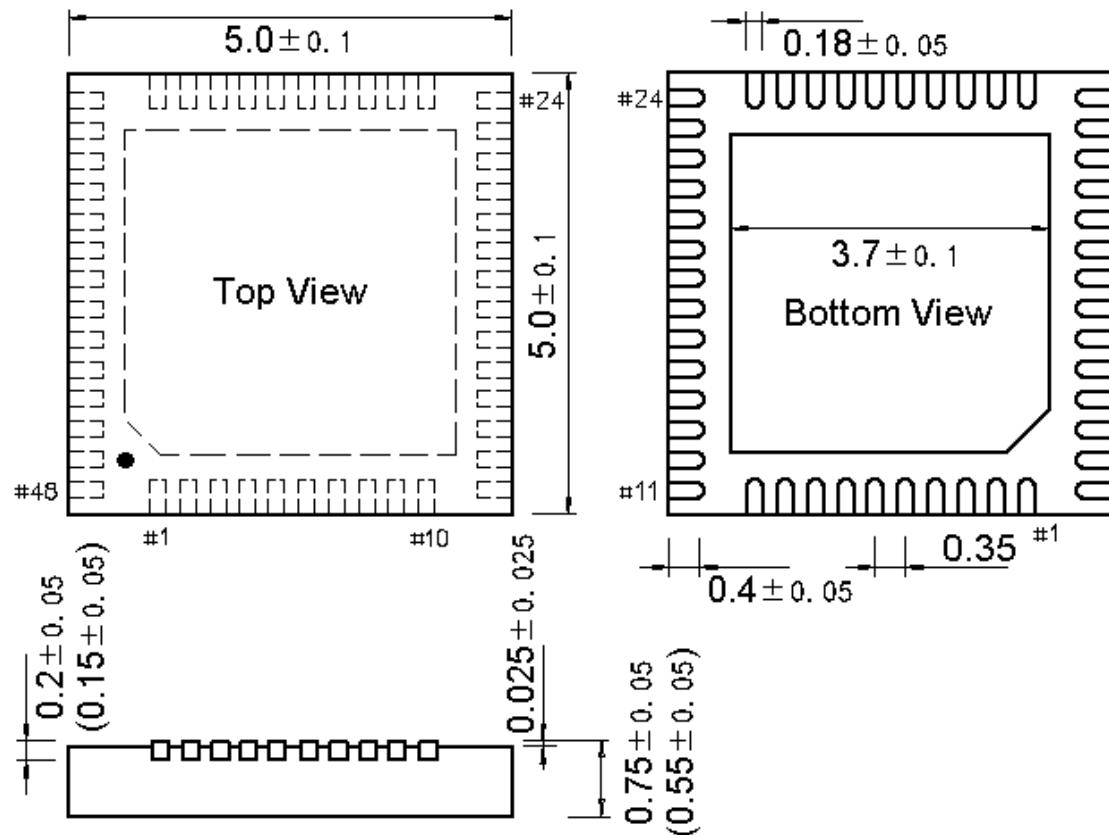
名称	参数说明	最小值	典型值	最大值	单位
V _{CC}	系统电源电压	2.4	3.3	3.6	V
I _{CC}	工作时的总电源电流		8.9		mA
V _{IL}	低电平输入电压	0		0.9	V
V _{IH}	高电平输入电压	2.0		V _{CC}	V
V _{OL}	低电平输出电压	0	0.3	0.4	V
V _{OH}	高电平输出电压	V _{CC} -0.4	V _{CC} -0.3	V _{CC}	V
I _{DN}	带下拉电阻输入端的输入电流	-90	-60	25	uA
I _{UP}	带上拉电阻输入端的输入电流	25	60	90	uA

10、封装信息

说明：尺寸标注的单位是 mm（毫米）。

引脚中心间距是标称值，没有误差，除此之外的尺寸误差不大于 $\pm 0.2\text{mm}$ 。

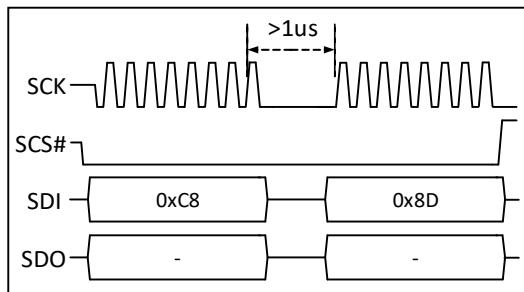
10.1 QFN48



11、应用

11.1 设置时钟

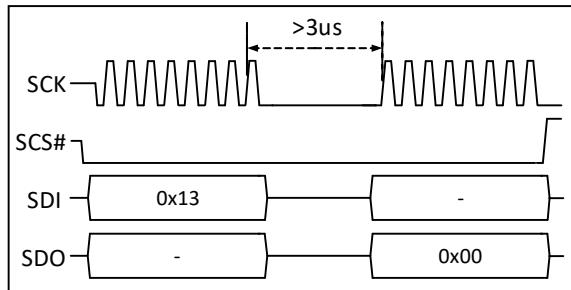
例：设置时钟为使用外部晶振，开启时钟倍频，分频系数为默认 13。



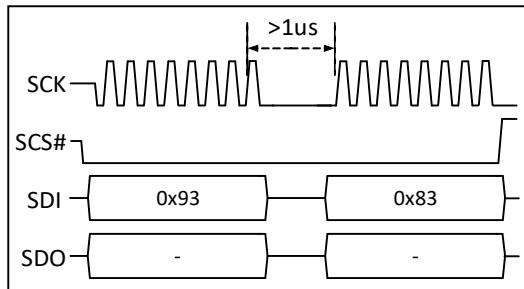
11.2 初始化串口

例：初始化串口 1 为 115200 波特率，8Bit 数据位，1Bit 停止位，无校验。

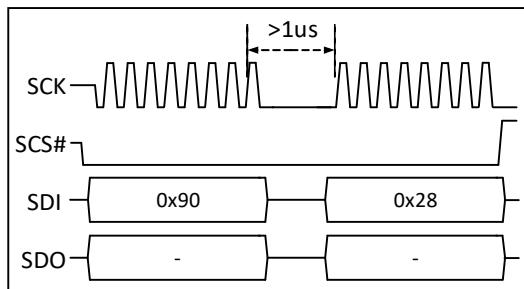
1)、读出串口 1 的 LCR 寄存器值，假设读出 0x00。

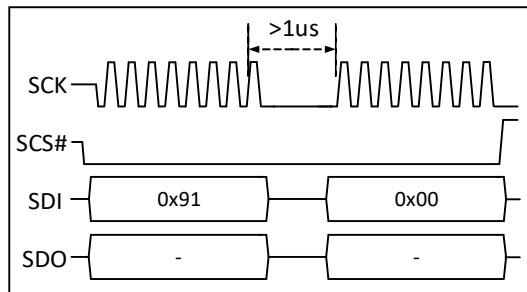


2)、写串口 1 的 LCR 寄存器值，设 DLM 位，设 8Bit 数据位，1Bit 停止位，无校验。

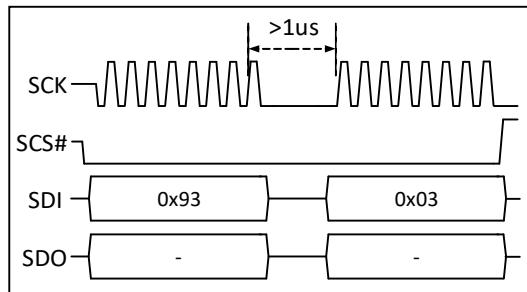


3)、设置串口 1 的波特率除数锁存器 DLL 和 DLM： $32000000 * 15 / 13 / 8 / 115200 \approx 40$ ，DLL 为 40，DLM 为 0。





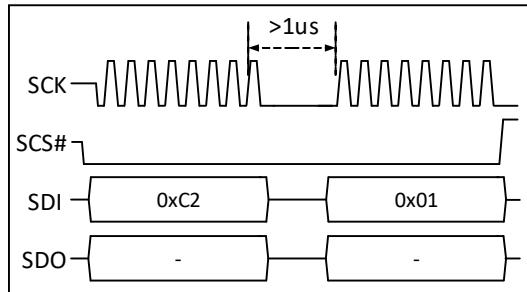
4)、关闭 DLAM。



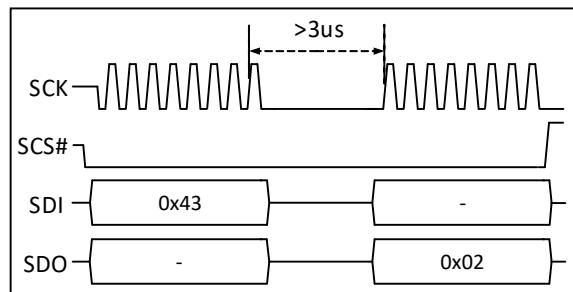
11.3 读取串口数据

例：串口 1 收到 2 字节数据 0x11、0x22。

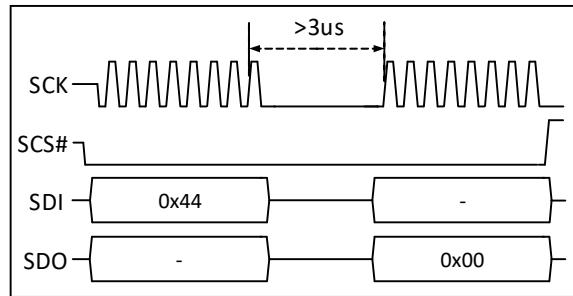
1)、设置读取接收 FIFO 计数器。



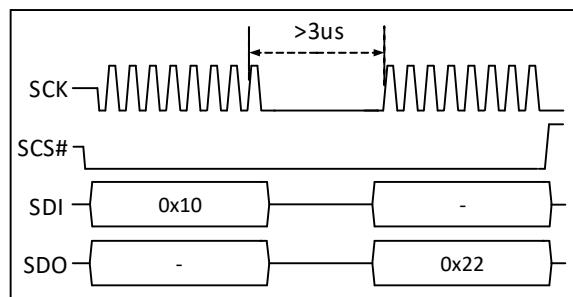
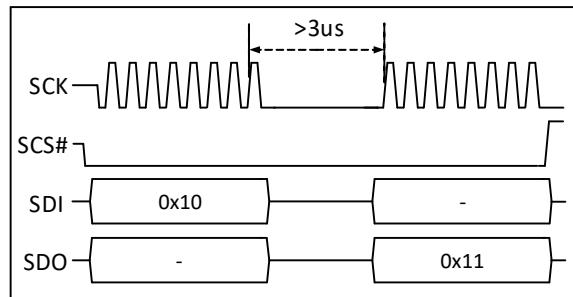
2)、读 FIFO 计数器低字节。



3)、读 FIFO 计数器高字节。



4)、读数据。



11.4 写串口数据

例：串口 1 发送 2 字节数据 0xAA、0xBB。

