SPI/IIC Extended Dual Serial Port Chip CH9432

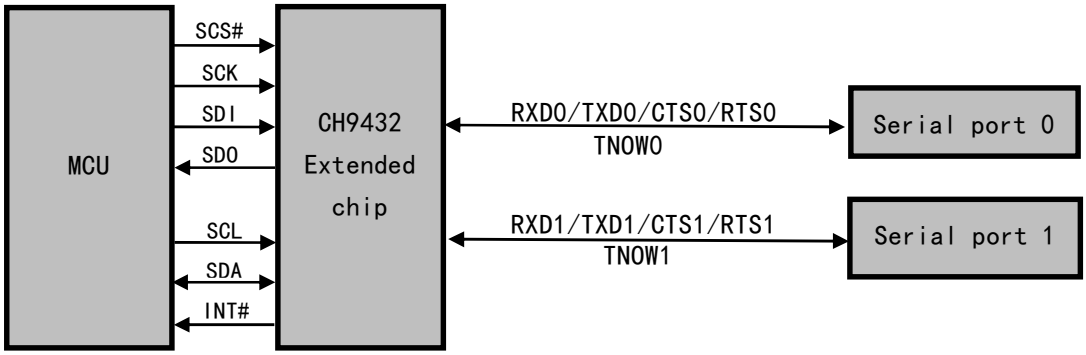
Datasheet

Version: 1.0

https://wch-ic.com

1. Overview

CH9432 is an SPI/IIC extended dual serial port expansion chip. The chip provides 2 sets of full-duplex 4-wire serial ports compatible with 16C550. It is used to expand UARTs for microcontrollers/embedded systems. It supports up to 6Mbps baud rate communication, supports up to 8-channel GPIO, provides half-duplex transceiver automatic switching pin TNOW, and supports RS485.



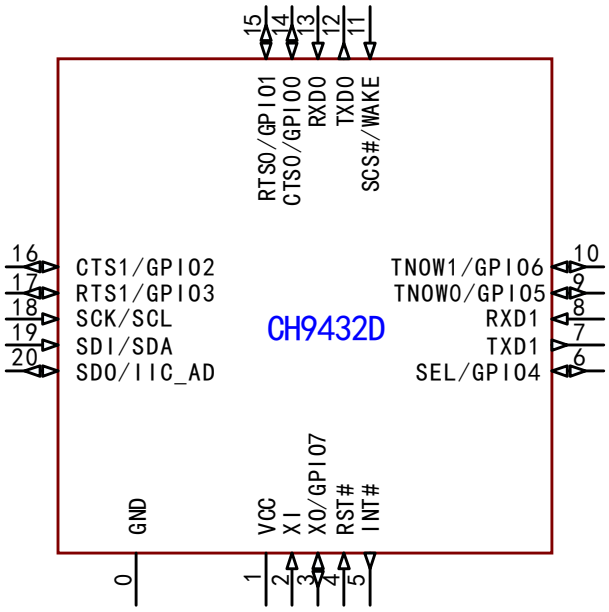
2. Features

- 2 completely independent UARTs, compatible with 16C550
- Support communication baud rate setting, up to 6Mbps
- Each direction of the serial port has independent FIFO buffer, transmitting 1536 bytes and receiving 2048 bytes.
- The serial port supports data bit setting and parity bit setting
- Provide half-duplex RS485 transceiver automatic switching pin, support RS485
- SPI up to 20Mbit/s
- Support multiple low-power modes and can be quickly woken up through the wake-up pin
- Provide configurable GPIO functionality
- Support internal clock or external 16M crystal to provide clock
- CH9432D is packaged in QFN20, lead-free package, RoHS compliant

3. Application Areas

- MCU/DSP/embedded system
- Industrial automation RS-485 communication
- Serial server, multi-serial card
- Communicates with Bluetooth, 4G, Wi-Fi and other serial port modules to achieve wireless transmission

4. Pinouts



Package form	Body size	Pin pitch		Package description	Order model
QFN20	3*3mm	0.4mm	15.8mil	Quad Flat No-lead Package	CH9432D

5. Pin Definitions

Table 5-1 CH9432D Pin definitions

Pin No.	Pin name(2)	Pin type(1)	Pin description
0	GND	P	Power ground.
1	V_{CC}	P	The chip power input requires an external power decoupling capacitor of no less than 0.1uF.
2	XI	A	XI: Inverting input terminal of high-frequency oscillator.
3	XO /GPIO7	I/O/A	X0: The inverting output terminal of the high-frequency oscillator. GPIO7: General-purpose bidirectional digital I/0 pin.
4	RST#	I	Chip reset pin, active low.
5	INT#	O	INT#: Interrupt output pin, active low.
6	SEL(3) /GPIO4	I/O	SEL: Main interface mode selection. GPIO4: General-purpose bidirectional digital I/0 pin.
7	TXD1	O	TXD1: Serial data output of UART1.
8	RXD1	I	RXD1: Serial data input of UART1.
9	TNOW0 /GPIO5	I/O (FT)	TNOW0: RS485 transceiver switching control pin of UART0. GPIO5: General-purpose bidirectional digital I/0 pin.
10	TNOW1 /GPIO6	I/O (FT)	TNOW1: RS485 transceiver switching control pin of UART1. GPIO6: General-purpose bidirectional digital I/0 pin.
11	SCS#	I(FT)	SCS#: SPI chip select input, active low.

	/WAKE		WAKE: Sleep wake-up pin.
12	TXD0	O(FT)	TXD0: Serial data output of UART0.
13	RXD0	I(FT)	RXD0: Serial data input of UART0.
14(5)	CTS0 /GPIO0	I/O (FT)	CTS0: MODEM contact input signal of UART0, clear to send, active low. GPIO0: General-purpose bidirectional digital I/0 pin.
15(5)	RTS0 /GPIO1	I/O (FT)	RTS0: MODEM contact output signal of UART0, request to send, active low. GPIO1: General-purpose bidirectional digital I/0 pin.
16	CTS1 /GPIO2	I/O (FT)	CTS1: MODEM contact input signal of UART1, clear to send, active low. GPIO2: General-purpose bidirectional digital I/0 pin.
17	RTS1 /GPIO3	I/O (FT)	RTS1: MODEM contact output signal of UART1, request to send, active low. GPIO3: General-purpose bidirectional digital I/0 pin.
18	SCK /SCL	I(FT)	SCK: SPI serial clock input. SCL: IIC clock pin.
19	SDI /SDA	I(FT)	SDI: SPI serial data input. SDA: IIC data pin.
20	SDO /IIC_AD	I/O (FT)	SDO: SPI serial data output. IIC_AD(4): IIC address setting bit.

Note 1: Pin type abbreviations:

P: Power pin; I: Input pin; O: Output pin; A: Analog pin; N: Empty pin. FT = Tolerant 5V

Note 2: After the reset is completed, the TXDx pin of the CH9432D chip is set to push-pull output by default, and other unspecified function pins are in floating mode. After the command sets the relevant pin enable and the I/O function is turned on, the pin will be automatically set to the corresponding state.

Note 3: After the SEL chip is reset, it is set to the pull-up input mode, which is used to select the communication interface with the master control. The high level selects the SPI interface or the low level selects the IIC interface to communicate with the master control. After the level determination is completed, the pull-up is automatically canceled and set to floating mode.

Note 4: When SEL selects the I2C interface, IIC_AD is automatically set to a pull-up input to determine the IIC address. After the level determination is completed, the pull-up is automatically canceled and set to floating mode.

Note 5: The PIN14 pin is set to the pull-up input mode during the chip reset, the PIN15 pin is set to the pull-down input mode during the chip reset, and is set to the floating input mode after the chip reset is completed.

6. Clock Configuration

CH9342 uses the chip's internal clock by default, and generates a 96MHz serial port reference clock after frequency multiplication. If the application requires a more accurate clock, an external crystal can be connected to the chip's internal clock oscillator to provide a 16MHz input clock. The main control can then turn on the external clock through register settings. The chip will automatically switch from the internal clock to the external clock, and generate a 96MHz serial port reference clock through frequency multiplication. The clock switching needs to be delayed for a period of time before other operations can be performed.

7. Register

7.1 Serial Register

The CH9432 chip provides 2 independent serial port modules, each using independent 8-byte registers for configuration. The addresses are: 00H-07H is serial port 0, 10H-17H is serial port 1. The serial port register is compatible with the industry standard 16C550 or 16C750 and has been enhanced. In the table, DLAB refers to bit 7 of the LCR register; X indicates that the DLAB value is irrelevant; RO indicates that the register is read-only; WO indicates that the register is write-only; and R/W indicates that the register is read-write.

Table 7-1 Serial register

Address	DLAB	R/W	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	R0	RBR	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	W0	THR	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 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	0	PARMODE1	PARMODE0	PAREN	STOPBIT	WORDSZ1	WORDSZ0
4	X	R/W	MCR	0	0	AFE	0	0	0	RTS	0
5	X	R0	LSR	RFIFOERR	TEMT	THRE	0	FRAMEERR	PARERR	OVERR	DATARDY
6	X	R0	MSR	0	0	0	CTS	0	0	0	△CTS
7	X	R/W	SCR	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	1	R/W	DLL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	1	R/W	DLM	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8

Note: If the chip does not have a corresponding functional pin, it will not possess this register function.

The following table shows the default values of the serial port registers after power-on reset or serial port soft reset.

Table 7-2 Default values

Register name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 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	0	0	0	0	0	0	0	0
SCR	Keep	Keep	Keep	Keep	Keep	Keep	Keep	Keep
FIFO	Reset, including transmit FIFO and receive FIFO							
TSR	Reset, TSR is the serial port transmit shift register							

RSR	Reset, RSR is the serial port receive shift register
Others	Undefined

RBR: Receive buffer register. If the DATARDY bit of LSR is 1, the received data can be read from this register. If FIFOEN is 1, the data received from the serial port shift register RSR is first stored in the receive FIFO and then read out through this register.

THR: Transmit holding register, including transmit FIFO, used to write data to be sent. If FIFOEN is 1, the written data is first stored in the transmit FIFO, and then output one by one through the transmit shift register TSR.

IER: Interrupt enable register, including enhanced function control bits and serial port interrupt enable.

RESET: When this bit is set to 1, the serial port is soft-reset; this bit automatically clears the register to zero, eliminating the need for software-based clearing.

LOWPOWER: When this bit is set to 1, the serial port's internal reference clock is disabled, causing the port to enter a low-power state.

IEMODEM: When this bit is set to 1, interrupts are enabled for changes in the modem input status.

IELINES: When this bit is set to 1, it enables the receive line status interrupt.

IETHRE: When this bit is set to 1, it enables the transmit hold register out-of-range interrupt.

IERECV: When this bit is set to 1, it enables the data received interrupt.

IIR: Interrupt identification register, used to analyze interrupt sources and handle them.

FIFOENS: This bit is the FIFO enable status. If it is 1, it means that the FIFO has been enabled.

Table 7-3 IIR register bits

IIR register bits				Priority	Interrupt type	Interrupt source	Clear interrupt method
IID3	IID2	IID1	NOINT				
0	0	0	1	None	No interrupt generated	No interrupt	-
0	1	1	0	1	Receive line status	OVERR, PARERR, FRAMEERR	Read LSR
0	1	0	0	2	Receive data is available	The number of bytes received reaches the FIFO trigger point	Read RBR
1	1	0	0	2	Receiving data timeout	The next data has not been received for more than 4 data times.	Read RBR
0	0	1	0	3	THR register empty	The transmit holding register is empty and IETHRE changes from 0 to 1 to re-enable the interrupt.	Read IIR Or write THR
0	0	0	0	4	MODEM input changes	△CTS	Read MSR

FCR: First-in first-out buffer FIFO control register, used to enable and reset the FIFO.

RECVTG1 and RECVTG0: Set the interrupt of the receive FIFO and the trigger point of hardware flow

control. 00 corresponds to 256 bytes, that is, receiving 256 bytes will generate an interrupt available for receiving data, and automatically invalidate the RTS pin when hardware flow control is enabled. 01 corresponds to 512 bytes, 10 corresponds to 1024 bytes, and 11 corresponds to 1280 bytes.

TFIFORST: If this bit is set to 1, the data in the transmit FIFO (excluding TSR) will be cleared. This bit can be cleared to 0 automatically and does not need to be cleared by software.

RFIFORST: If this bit is set to 1, the data in the receiving FIFO (excluding RSR) will be cleared. This bit can be cleared to 0 automatically and does not need to be cleared by software.

FIFOEN: If this bit is 1, FIFO is enabled; 'FIFOEN' is set to 1 to enable serial port transmission, and to 0, the serial port is closed. Single-byte transmission mode is not supported.

LCR: Line control register, used to control the format of serial communication.

DLAB: This bit is the divisor latch access enable. When it is 1, DLL and DLM can be accessed, and when it is 0, RBR/THR/IER can be accessed.

PARMODE1 and PARMODE0: Set the parity bit format when PAREN is 1: 00 means odd parity, 01 means even parity.

PAREN: If this bit is 1, it allows the parity bit to be generated when transmitting and checked when receiving. If it is 0, there is no parity bit.

STOPBIT: If this bit is 1, it means 2 stop bits, if it is 0, it means one stop bit.

WORDSZ1 and WORDSZ0: Set the word length; 00 indicates 9-bit mode, 11 indicates 8-bit mode, and other values are invalid. The 9-bit mode is used for the parity check function.

MCR: MODEM control register, used to control MODEM output.

AFE: Setting this bit to 1 allows CTS and RTS hardware automatic flow control. If AFE is 1, the serial port will continue to send the next data only when it detects that the CTS pin input is valid (active low), otherwise the serial port transmission will be suspended. If AFE is 1 and RTS is 1, then when the receiving FIFO is empty, the serial port will automatically validate the RTS pin (active low). Until the number of bytes received reaches the trigger point of the FIFO, the serial port will automatically invalidate the RTS pin, and can validate the RTS pin again when the receiving FIFO is empty. Using hardware automatic control, you can connect your own CTS pin to the other party's RTS pin, and connect your own RTS pin to the other party's CTS pin.

RTS: If this bit is 1, the RTS pin output is valid (active low), otherwise the RTS pin output is invalid.

LSR: Line status register, used to analyze serial port status in query mode.

RFIFOERR: If this bit is 1, it indicates that there is at least one PARERR and FRAMEERR error in the receive FIFO.

TEMT: If this bit is 1, it indicates that the transmit holding register THR and the transmit shift register TSR are all empty.

THRE: If this bit is 1, it indicates that the transmit holding register THR is empty.

FRAMEERR: When this bit is 1, it indicates that there is a framing error in the data currently read from the receive FIFO, usually due to the lack of a valid stop bit.

PARERR: If this bit is 1, it indicates a parity error in the data being read from the receive FIFO.

OVERR: If this bit is 1, it indicates a receive FIFO buffer overflow.

DATARDY: If this bit is 1, it indicates that there is received data in the receive FIFO; this bit is automatically cleared to 0 after all data in the FIFO has been read.

MSR: MODEM status register, used to query MODEM status.

CTS: This bit is the inverse of the CTS pin. If it is 1, it means that the CTS pin is valid (active low).

△CTS: If this bit is 1, it indicates that the input status of the CTS pin has changed.

SCR: User-definable register.

DLL and DLM: Baud rate divisor latch, DLL is the low byte, DLM is the high byte, the 16-bit divisor composed of the two is used for the serial port baud rate generator composed of a 16-bit counter.

The CH9432 defines DLM and DLL as a 16-bit data field, with the upper 12 bits representing the integer division (DIV_M:[15:4]), with the lower 4 bits representing the fractional division (DIV_F:[3:0]). The communication baud rate is calculated as: Serial Port Reference Clock/(16 * (DIV_M + DIV_F/16)). The division values derived from this formula may have some deviation, so it is recommended to select values that are as close as possible.

7.2 Interface Register

Table 7-4 Serial register

Address	Name	Function	Bit	Access	Bit definition	Default value
41H	R8_TNOW_CTRL_CFG	TNOW function settings	[7:6]	R0	Reserved	00b
			[5:4]	R/W	[4]: TNOW0 signal level polarity configuration 0: Normal mode; 1: Level inversion. [5]: The definition of TNOW1 signal level polarity configuration value is the same as above.	00b
			[3:2]	R0	Reserved	00b
			[1:0]	R/W	[0]: TNOW0 pin function control 0: Disable TNOW function; 1: Enable TNOW function. [1]: TNOW1 pin function control, the value definition is the same as above.	00b
42H	R8_FIFO_CTRL	FIFO counter settings	[7:5]	R0	Reserved	000b
			4	R/W	Transceiver buffer control 0: Receive FIFO; 1: Transmit FIFO.	0
			[3:1]	RO	Reserved	000b
			0	R/W	Serial port selection 0: Select serial port 0; 1: Select serial port 1;	0

					Returns the currently selected serial number when reading.	
43H	R8_FIFO_CNT_L	FIFO counter	[7:0]	R0	FIFO counter lower 8 bits	XXh
44H	R8_FIFO_CNT_H	FIFO counter	[7:0]	R0	FIFO counter high 8 bits	XXh
45H	IO_SEL_FUN_CFG	Pin function multiplexing enable	[7:0]	R/W	Control command	00h
			[7:0]	R/W	Communication data 0	00h
			[7:0]	R/W	Communication data 1	00h
			[7:0]	R/W	Communication data 2	00h

R8_TNOW_CTRL_CFG: Set the TNOW function opening and polarity. The polarity is default (regular mode). When sending data, the corresponding TNOW pin is pulled high, and TNOW is pulled low after sending.

R8_FIFO_CTRL: FIFO counter setting. After setting the FIFO direction and serial port, read R8_FIFO_CNT_L and R8_FIFO_CNT_H to obtain the corresponding FIFO counter value. Among them, the read serial port sending FIFO counter value is the current remaining FIFO quantity, and the master control can fill in the maximum amount of sending data according to this value; the reading receiving FIFO counter value is the current received data quantity, and the master control can read all the serial port data at one time according to this value.

IO_SEL_FUN_CFG: Chip pin function multiplexing setting. During communication, 4-byte data must be read and written completely, otherwise the setting will fail. For details, see the pin function enabling instructions.

7.3 Clock Power Register

Table 7-5 Clock power register

Address	Name	Function	Bit	Access	Bit definition	Default value
48H	R8_CLK_CTRL_CFG	Clock settings, only switch clock source	[7:6]	R/W	System clock source mode selection: x0: Internal clock; x1: Use external crystal oscillator.	00b
			[5:0]	R0	Reserved.	00000b
4AH	R8_SLEEP_MOD_CFG	Sleep function settings	[7:3]	R0	Reserved.	00000b
			[2:0]	R/W	Chip low power settings: 0: Do not enter low power consumption state; 1: IDLE state; 2: SLEEP state; 3: SHUT state; 4-7: Reserved.	000b

7.4 GPIO Register

Table 7-6 CH9432 GPIO register

Address	Name	Function	Bit	Access	Bit definition	Default value

50H	R8_GPIO_FUNC_EN_0	GPIO7-0 enable	[7:0]	R/W	[7:0] Corresponds to GPIO 7-0 enable control: 0: Disables the GPIO function; 1: Enables the GPIO function.	00h
54H	R8_GPIO_DIR_MOD_0	GPIO1-0 mode settings	[7:0]	R/W	[7:4] corresponds to the GPIO1 mode settings; [3:0] corresponds to the GPIO0 mode settings. Definition of each 4-bit group: [3:2]: IO function configuration: In input mode: 01: Floating input; 10: Input with pull-up/pull-down resistors. In output mode: 00: Push-pull output; 01: Open-drain output. [1:0]: Mode settings for this I/O: 00: Input mode; 11: Output mode.	00h
55H	R8_GPIO_DIR_MOD_1	GPIO3-2 mode settings	[7:0]	R/W	[7:4] corresponds to the GPIO3 mode settings; [3:0] corresponds to the GPIO2 mode settings; The values are defined as above.	00h
56H	R8_GPIO_DIR_MOD_2	GPIO5-4 mode settings	[7:0]	R/W	[7:4] corresponds to the GPIO5 mode settings; [3:0] corresponds to the GPIO4 mode settings; The values are defined as above.	00h
57H	R8_GPIO_DIR_MOD_3	GPIO7-6 mode settings	[7:0]	R/W	[7:4] corresponds to the GPIO7 mode settings; [3:0] corresponds to the GPIO6 mode settings; The values are defined as above.	00h
5CH	R8_GPIO_SET_0	GPIO7-0 set	[7:0]	R/W	[7:0] Corresponds to the set control for GPIO7-0; When a bit is set to 1, the I/O pin outputs a high level; setting it to 0 has no effect. This also affects the 'GPIO_PIN_VAL' register.	00h
5EH	R8_GPIO_RESET_0	GPIO7-0 reset	[7:0]	R/W	[7:0] Corresponds to the GPIO7-0 reset control;	00h

					When a 1 is written to the corresponding bit, the IO output goes low; Writing a 0 has no effect and also affects the "GPIO_PIN_VAL" register.	
60H	R8_GPIO_PIN_VAL_0	GPIO7-0 Level output, level reading and pull-up and pull-down settings	[7:0]	R/W	[7:0] Corresponds to the voltage level of GPIO pins 7–0. Writing to the register: When GPIO is set to output: 1: Output a high voltage; 0: Output a low voltage. When GPIO is set to input and pull-up/pull-down resistors are enabled: 1: Enable the pull-up resistor; 0: Enable the pull-down resistor. Reading from the register: When GPIO is set to input: The voltage level of the pin.	00h

Note: Chips of different packages or models do not have this register function if they do not have corresponding functional pins.

7.5 Extended Register

Table 7-7 Extended register

Address	Name	Function	Bit	Access	Bit definition	Default value
65H	CHIP_VER_ADD	Current firmware version	[7:0]	RO	VER[0]: The lower 8 digits of the current version number.	XXh
			[7:0]	RO	VER[1]: The higher 8 digits of the current version number.	XXh
			[7:0]	RO	VER[2]: Version information verification.	XXh
			[7:0]	RO	VER[3]: End character.	9Eh

Note: The version number is only used for identification. The functions of the higher version chip are compatible with the lower version. The main control MCU cannot be compared for a certain version number.

8. Function Description

8.1 Interrupts and Queries

The INT# pin of the CH9432 chip is an interrupt request output pin. The INT# pin is in floating mode after the chip is powered on. After being enabled, it is in push-pull output mode and is active at low level. Since the two serial ports share an interrupt pin output, when there is an interrupt valid signal, the main control MCU needs to query the interrupt status of all serial ports to analyze which serial port has an interrupt request.

If the serial port has multiple interrupt request outputs, when the main control MCU reads a valid interrupt status, CH9432 will temporarily pull the interrupt pin high, and then pull it low. The main control MCU can query and judge, and directly query the serial port interrupt status as long as the interrupt pin is low level.

8.2 Serial Port Characteristics

The serial data of the CH9432 serial port includes 1 low-level start bit, 8 data bits, 1 or 2 high-level stop bits, and supports odd or even parity. Supports common communication baud rates: 3600, 4800, 9600, 14400, 19200, 28800, 33600, 38400, 56000, 57600, 76800, 115200, 128000, 153600, 230400, 460800, 921600, 250K, 500K, 1M, 1.5M, 2M, 3M, 4M, 6M.

CH9432 ensures the reliability of multi-serial communication through high-precision clock design. When using a 16MHz crystal oscillator clock source and working at the above common baud rate, the baud rate error of the transmitter is less than ±1%; at the same time, it allows the baud rate error of the opposite end to be ±2%.

8.3 Serial Port Operation

The operation of the serial port provides IIR and LSR registers to query the transmitting and receiving status of the serial port. CH9432 also extends a FIFO register to directly obtain the amount of data sent and received by the current chip's serial port.

When the serial port interrupt queries to receive data, you can first read the receiving FIFO length of the current serial port, and then directly operate the RBR register to read all data. The main control MCU can also simplify the processing and directly query the current number of received data in the FIFO regularly, and then operate the RBR register to read the data according to the number of received FIFOs.

To send data, you can query the empty status of the transmit buffer of the IIR and LSR registers, and then operate the THR register to send data through the serial port. The main control MCU can also simplify the processing and use the query FIFO size for data transmission. The value of the query send FIFO data length is the remaining FIFO size. The main control MCU can use this as the size to fill the send data into the THR register in sequence for data transmission.

To enable the serial port flow control function, set the AFE bit to 1, and CH9432 will automatically perform hardware flow control. The chip will automatically operate the flow control pin according to the FIFO size. After enabling automatic flow control, the chip serial port will continuously send data when CTS is valid. When the CTS pin is invalid, CH9432 will immediately stop sending. RTS automatically expires after the receiving FIFO

reaches the set number of flow control bytes. Note that there is a certain delay in RTS failure control.

8.4 RS485 Switch Pin TNOW

The CH9432 serial port provides an RS485 switching pin (TNOW), which shares its pin function with other features. After the chip has powered up, the host controller can enable the corresponding TNOW pin and then configure the TNOW setting register to activate the TNOW function. TNOW also supports polarity adjustment to accommodate different polarity scenarios.

8.5 GPIO Function

CH9432 supports function pins to be multiplexed as GPIO functions, and CH9432D supports up to 8 channels. GPIO can set direction, pull-up resistor and pull-down resistor configuration. When setting the GPIO function, you need to pay attention to the original function mode of the IO and the setting sequence of the IO registers to prevent 'jitter' in the IO.

8.6 Low-power Mode

CH9432 supports 3 sleep modes, namely IDLE mode when the chip stops running, SLEEP mode when the clock is turned off, and SHUT power-down sleep mode. The wake-up pins of the SPI and IIC interfaces are both SCS# (WAKE) pins. They wake up on the falling edge. The master control needs to pay attention to the level status to prevent leakage current.

After waking up in IDLE mode, several uS can communicate. After waking up from SLEEP mode, it takes about a few mS to wait for the power supply and clock to start, and then communication can be carried out. The chip in SHUT mode will be reset again after waking up. At this time, the main control needs to be reinitialized to continue using it.

8.7 SPI Serial Interface

The SPI synchronous serial interface signal line includes: SPI chip select input pin SCS#, serial clock input pin SCK, serial data input pin SDI and serial data output pin SDO. SPI data bit order: high-order bit first. The format of the sent data is that the first byte is the operation address, the second byte is the written data or the read data, the highest bit of the operation address is the operation bit, if the operation bit is 1, it means writing data, and if it is 0, it means reading data.

When CH9432 writes register data, from the last clock edge when the address byte transmission is completed to the last clock edge when the first data byte transmission is completed, the entire time window needs to be greater than 2uS; after the data SCS# is released, a delay of 3uS is required before the next operation can be performed. When reading register data, the address and data need to be delayed by 3uS, that is, after the address is sent, there is a delay of 3uS for data reading. After the data is read and SCS# is released, a delay of 3uS is required before the next operation can be performed.

CH9432 supports continuous reading and writing of serial port data registers by default. Transmission method: After transmitting the register address, there is a delay of 3uS, and the subsequent data part can be read and written continuously until the cache ends. If a buffer overflow occurs, such as serial port reception, the read data

needs to be read in packets, so that the buffer size (2048 bytes) is not exceeded for packet reading.

8.8 Version Number

Reading the version number requires reading 4 bytes of version information, where VER[0] and VER[1] are version information, and the parsing format is BCD code. The high 4 bits of the VER[1] field are reserved information (fixed to 0), VER[2] is the version information verification field, the value is (VER[0]+VER[1]), VER[3] is the end field, and the value is fixed at 0x9E. When reading, the main control MCU needs to perform verification field analysis to determine whether the version number is legal. If the verification field is parsed successfully, it means that the chip has version information. The version number is only used for identification. The functions of higher version chips are compatible with lower versions. The main control MCU should not be locked to a single version.

8.9 Pin Function Enable Description

After the CH9432 chip is powered on, the TXDx pin defaults to push-pull output mode, and the other functional pins default to floating mode. The pins, such as serial port pins, crystal oscillator pins, INT# pins, etc., need to be set through the main control interface before they are turned on. Set the pin function through the IO_SEL_FUN_CFG register. The reading and writing of the pin multiplexing register are processed according to 4 bytes. The first byte is the command code, and the following three bytes are data; the register sets the multiplexing pin and the default pin. When reading and writing the register, you need to follow the following steps: perform operations-wait for response, two steps, in which the data content is the first Three bytes are status flags, which need to be set to 0xA5 when the master control writes. After the chip operation is completed, the third byte will be set to 0x5A. After the master control performs the operation, it is recommended to delay reading until the third byte is 0x5A, indicating the end of execution. If it is not read, you can retry the read.

During the initialization stage of CH9432, the main control needs to first open the IO according to the application, otherwise the IO function will be unusable.

Register setting commands and data:

Control command	Function	Data content, '[]' is a single-byte definition
0x01	Set multiplexing function enable	Master write: [Multiplexed pin address][Enable][0xA5] Read returns: [Multiplexed pin address] [no definition] [0x5A]
0x81	Read the corresponding multiplexed pin enable status	Master write: [Multiplexed pin address] [no definition] [0x5A] Read returns: [Multiplexed pin address][Enable][0xA5]
0x03	Set default pin enable	Master write: [Multiplexed pin address][Enable][0xA5] Read returns: [Multiplexed pin address] [no definition] [0x5A]
0x83	Read default pin enable status	Master write: [Multiplexed pin address] [no definition] [0x5A] Read returns: [Multiplexed pin address][Enable][0xA5]

Default pin address: Just turn it on when using it.

Default pin function	Default pin address	Default PIN number
UART0_TX_RX	1	PIN12, PIN13
UART1_TX_RX	2	PIN7, PIN8

HSE_XI_XO	3	PIN2 PIN3
CTS0	4	PIN14
CTS1	5	PIN16

Multiplexed pin address and corresponding default PIN number:

Default pin function	Default pin address	Default pin PIN number
INT#	1	PIN5
TNOW0	2	PIN9
TNOW1	3	PIN10
RTS0	4	PIN15
RTS1	5	PIN17
GPIO0	6	PIN14
GPIO1	7	PIN15
GPIO2	8	PIN16
GPIO3	9	PIN17
GPIO4	10	PIN6
GPIO5	11	PIN9
GPIO6	12	PIN10
GPIO7	13	PIN3

8.10 IIC Interface

CH9432 supports IIC interface. The 2-wire IIC bus includes data line SDA and serial clock line SCK. The pin is in open-drain mode and requires an external pull-up resistor. The rate supports 1MHz and 7-bit address. The lowest bit of the address is configured through the pin IIC_AD. The high-bit value of the address is 0x28. That is, the address value is 0x28|IIC_AD. During the power-on initialization phase, the chip selects the communication interface based on the current SEL pin. If the IIC interface is selected, the chip will initialize the IIC_AD pin and record the current level status. After the address setting is completed, the IIC_AD pin will be set to a floating state.

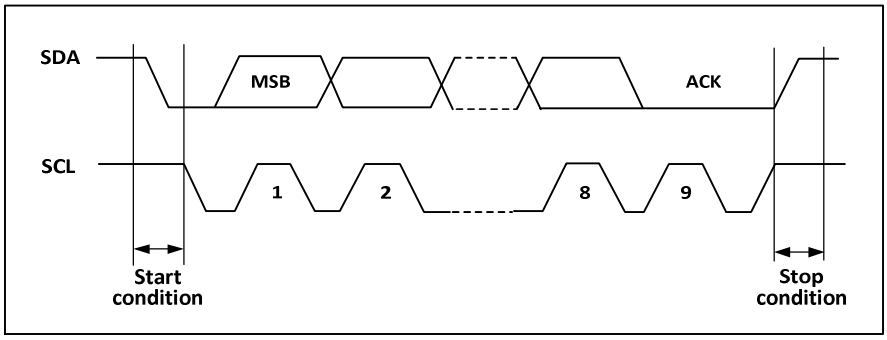
The address and data transmitted by IIC are high-order bit first, low-order bit last. The first byte after the start event is the address byte, the first 7 bits are the device target address, and the 8th bit (Bit0 in the table below) is the transmission message direction. 0 means that the master device writes data to the slave device, and 1 means that the master device reads data from the slave device.

Address byte content diagram:

bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	**1**	**0**	**1**	**0**	**0**	**IIC_AD**	**0/1**

IIC is a half-duplex bus, and a complete transmission includes start signal, data, response and end signals. The first byte is the IIC address byte, and the subsequent bytes are the transmission protocol. The chip matches according to the address, and the device address on the bus must be unique.

IIC transmission timing diagram:



IIC write register: After the IIC start event, first write the write direction IIC address byte, then write the register address byte, and finally write the corresponding data byte. After the data is written, the IIC end event is sent.
IIC write register flow diagram:

S	IIC address byte								A	Register address								A	N byte data	A	P
	0	1	0	1	0	0	AD	0		1	x	x	x	x	x	x	x		D0-Dn-1		

The N-byte data written in the register can be single byte or multiple bytes.

IIC read register: After the IIC start event, the write direction IIC address byte is first written, and then the register address byte is written. Send the IIC start event again, write the IIC address byte in the read direction, read the data, and send the IIC end event after the data reading is completed.
IIC read register flow diagram:

S	IIC address byte								A	Register address								A
	0	1	0	1	0	0	AD	0		0	x	x	x	x	x	x	x	

S	IIC address byte								A	Read N byte data	NA	P
	0	1	0	1	0	0	AD	1		D0-Dn-1		

The data read can be single byte or multiple bytes.

S: IIC start signal;
A: IIC acknowledge signal; an acknowledge signal is required after each byte is transmitted; the diagram omits the acknowledges between multiple bytes;
NA: No acknowledge signal for this IIC byte;
P: IIC stop signal;
AD: The IIC configuration address, i.e., IIC_AD;
The register address and data content transmitted by IIC are consistent with the SPI interface. The data content can refer to the SPI interface.
IIC transmission exception recovery: To prevent abnormal situations in the interface during IIC transmission, the chip will monitor the IIC transmission process. When an IIC transmission error occurs or the transmission is interrupted for more than 40mS, the chip will reset the transmission and wait for the next start signal and IIC address.

9. Parameters

9.1 Absolute Maximum Value

(Critical or exceeding the absolute maximum value will likely cause the chip to work improperly or even be damaged)

Name	Parameter	Min.	Max.	Unit
T_A	Ambient temperature during operation	-40	85	℃
T_S	Ambient temperature during storage	-40	125	℃
V_{CC}	Chip power voltage（V_{CC} to power, GND to ground）	-0.3	4.0	V
V_{IN}	Input voltage on FT (5V tolerant) pin	-0.3	5.5	V
	Voltage on other pins	-0.3	V_{CC}+0.3	V
$V_{ESD(HBM)}$	ESD electrostatic discharge voltage (HBM) of common I/O pins	4K		V

9.2 Electrical Parameters (Test condition: T_A=25℃, V_{DD}=3.3V)

Name	Parameter description	Min.	Typ.	Max.	Unit
V_{CC}	Chip power supply voltage	1.8	3.3	3.6	V
I_{CC}	Total supply current during operation		8.1		mA
V_{IL}	Standard I/O, low level input voltage	0		1.0	V
	FT I/O, low level input voltage	0		1.0	V
V_{IH}	Standard I/O, high level input voltage	2.0		3.3	V
	FT I/O, high level input voltage	1.9		5.0	V
V_{OL}	At 8mA current, low-level output voltage			0.4	V
V_{OH}	At 8mA current, high-level output voltage	2.9			V
R_{PU}	Pull-up equivalent resistance	30	40	50	KΩ
R_{PD}	Pull-down equivalent resistance	30	40	50	KΩ

9.3 Low-power Parameters (Test condition: T_A=25℃, V_{DD}=3.3V)

Name	Parameter description	Min.	Typ.	Max.	Unit
I_{CC1}	Low-power mode 1: IDLE SLEEP		4.6		mA
I_{CC2}	Low-power mode 2: SLEEP		4.5		uA
I_{CC3}	Low-power mode 3: SHUT SLEEP		0.4		uA

Note: In low-power consumption mode, attention should be paid to the level status and settings of I/O to avoid leakage current caused by floating. The above parameters are typical values.

9.4 SPI/IIC Serial Interface Related Parameters

Name	Parameter description	Min.	Typ.	Max.	Unit
f_{SCL}/t_{SCL}	IIC interface clock frequency			1	MHz

f_{SCK}/t_{SCK}	SPI interface clock frequency			20	MHz
t_{SD1}	When SPI writes registers (excluding RBR and THR registers), address and data time windows	2			uS
t_{SD2}	When SPI reads registers (excluding RBR and THR registers), the address and data time window	3			uS
t_{SD3}	Address and data time windows when SPI writes to FIFO (THR register)	3			uS
t_{SD4}	Address and data time windows when SPI reads FIFO (RBR register)	3			uS
t_{SD5}	Time interval between two SPI operation addresses	3			uS

Note: Since the maximum clock frequency of the IIC interface is 1MHz, there is no need to insert a delay between the register address and data when the IIC interface reads and writes registers and reads and writes FIFO (RBR or THR register).

Figure 9-1 IIC mode timing

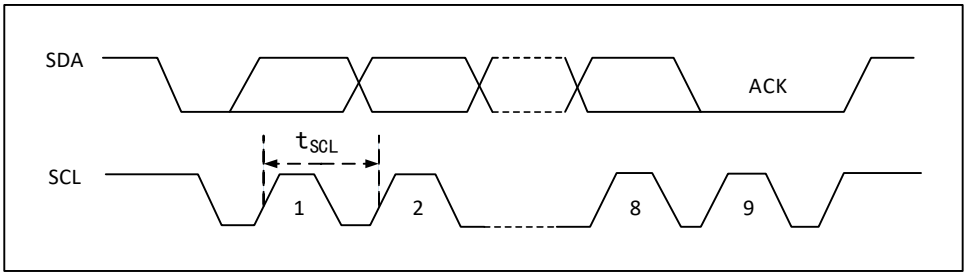


Figure 9-2 SPI mode write register

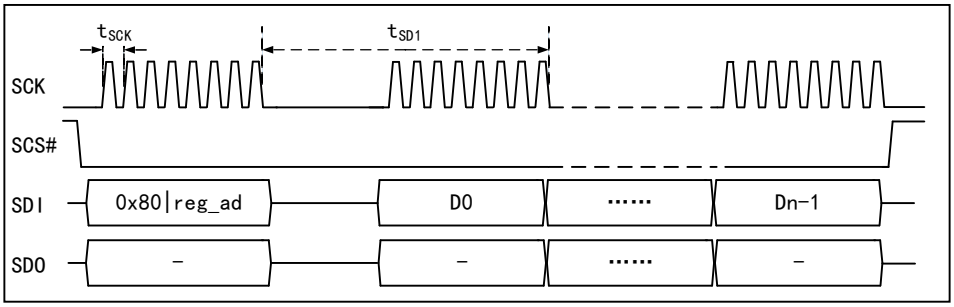


Figure 9-3 SPI mode read register

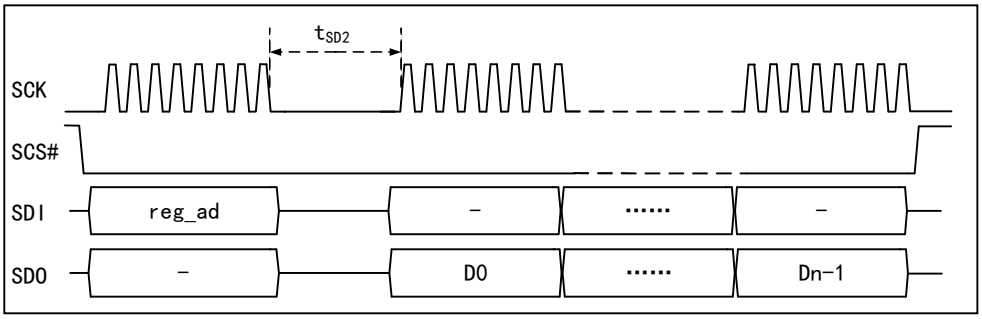


Figure 9-4 SPI mode write FIFO

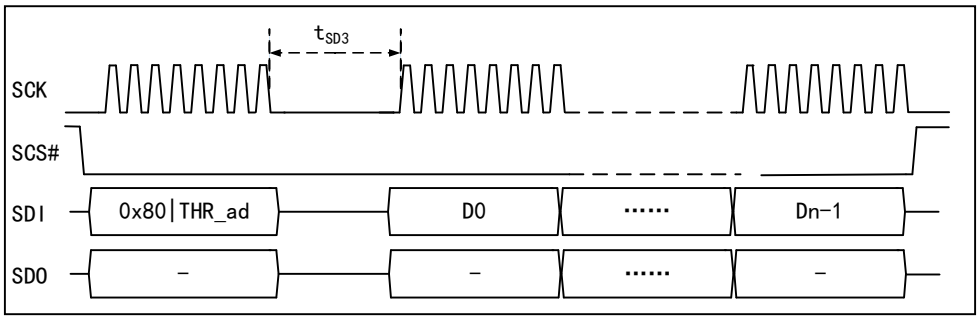


Figure 9-5 SPI mode read FIFO

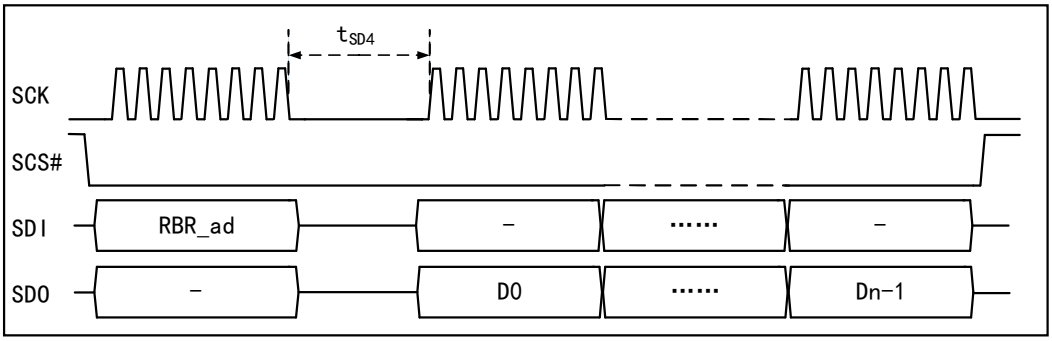
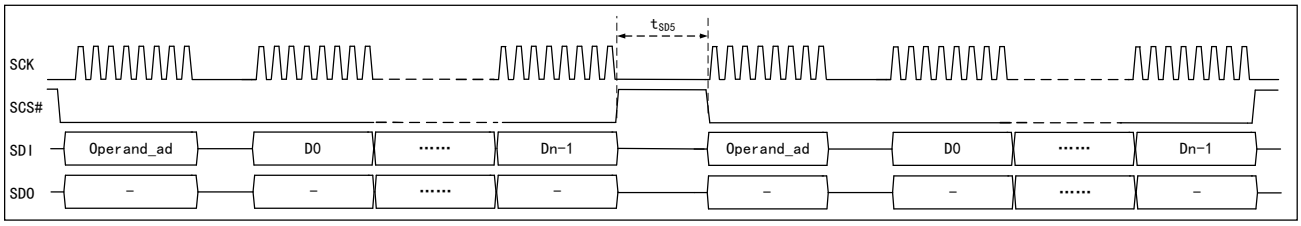


Figure 9-6 SPI mode continuous operation register

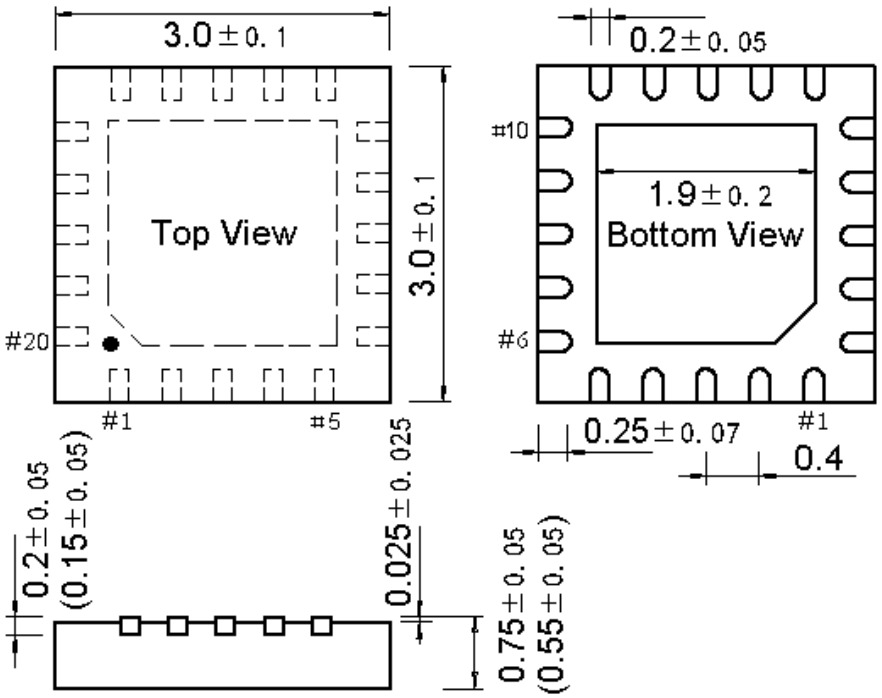


10. Package Information

Note: All dimensions are in millimeters.

The pin center spacing values are nominal values, with no error. Other than that, the dimensional error is not greater than the greater of ±0.2mm.

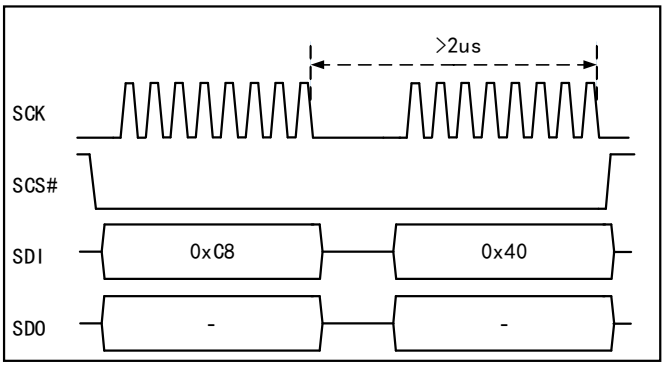
10.1 QFN20



11. Application

11.1 Set Clock

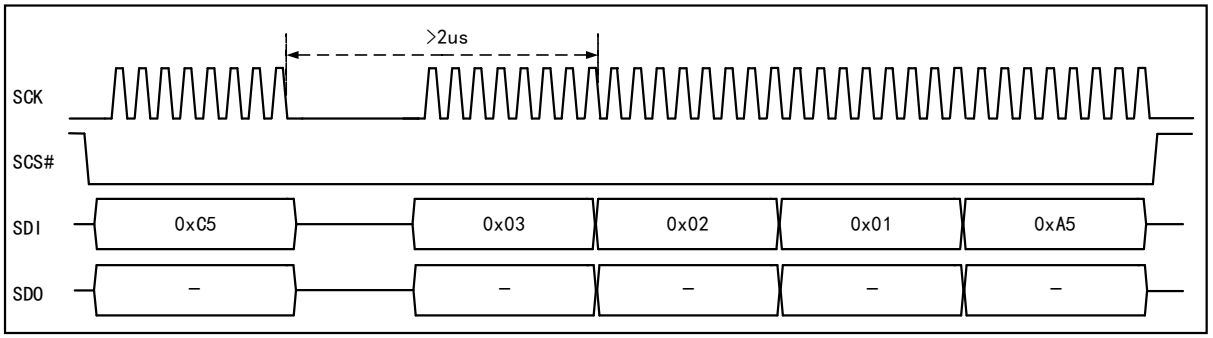
Example: Set the CH9432 system clock source to an external crystal oscillator.

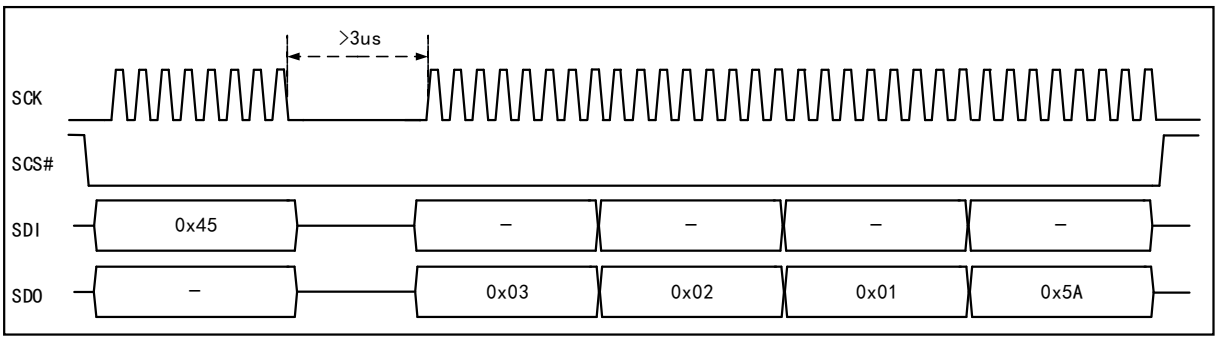


11.2 Initialize Serial Port

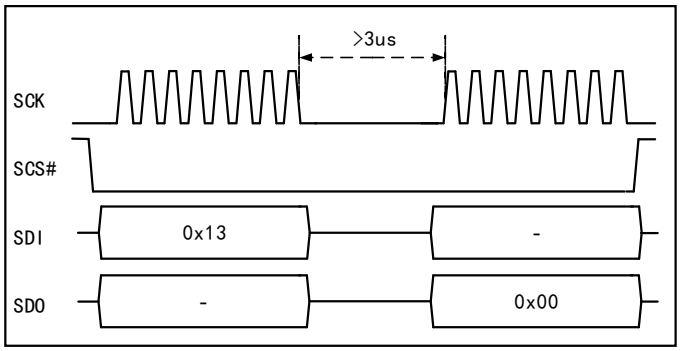
Example: CH9432 initializes serial port 1 with a baud rate of 115200, 8 bit data bits, 1 bit stop bit, and no parity.

1). Turn on the pin function of serial port 1

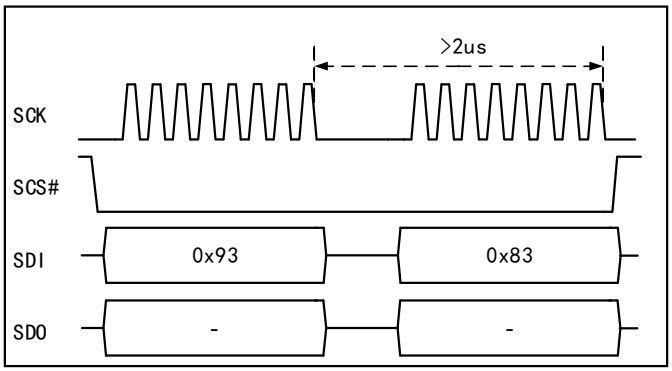




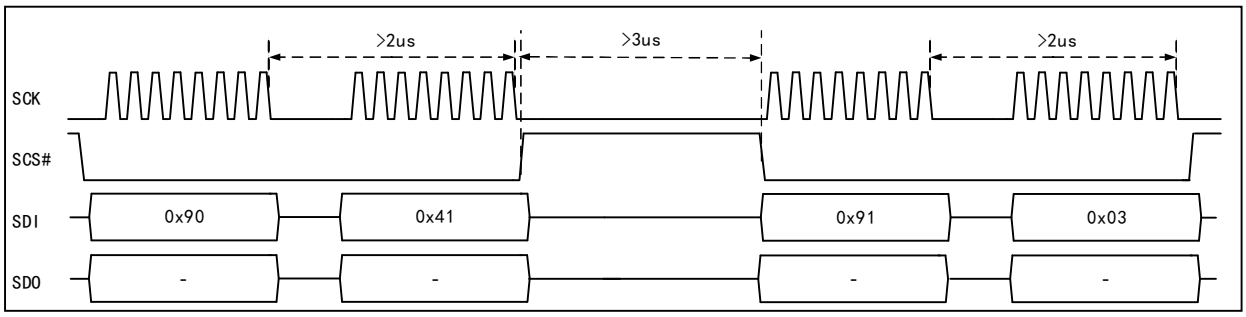
2). Read the LCR register value of serial port 1. Suppose it reads 0x00.



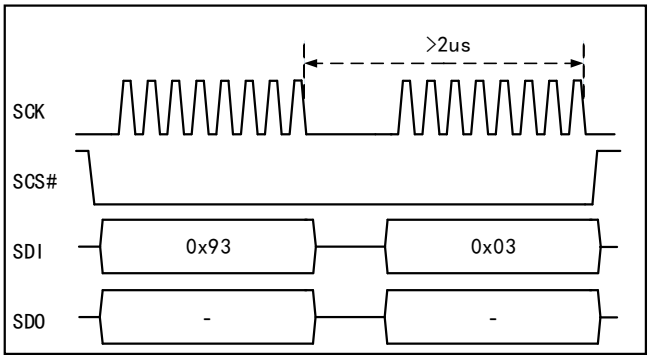
3). Write the LCR register value of serial port 1, set the DLAB bit, set the 8-bit data bit, 1-bit stop bit, and no check



4). Set the baud rate divisor latches DLL and DLM of serial port 1. When the baud rate is 115200, DLL is 0x41 and DLM is 0x03.



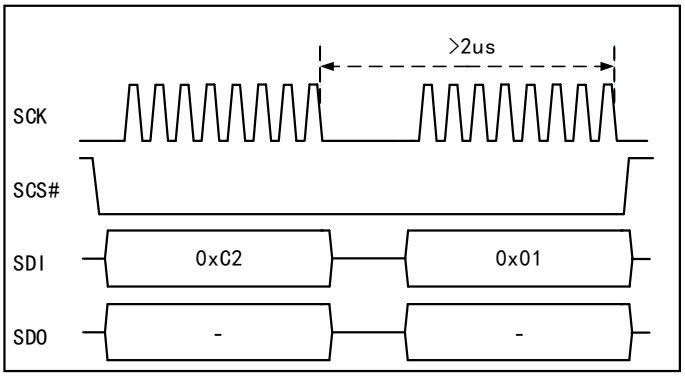
5). Disable DLAB



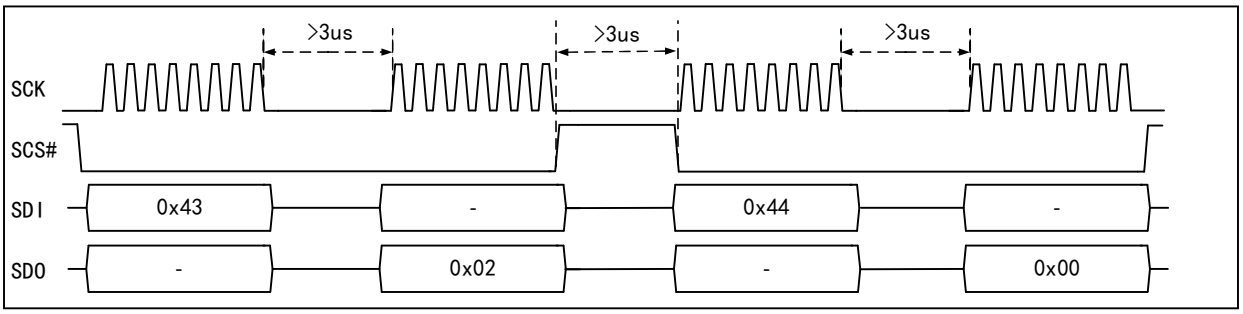
11.3 Read Serial Port Data

Example: CH9432 serial port 1 receives 2-byte data 0x11, 0x22.

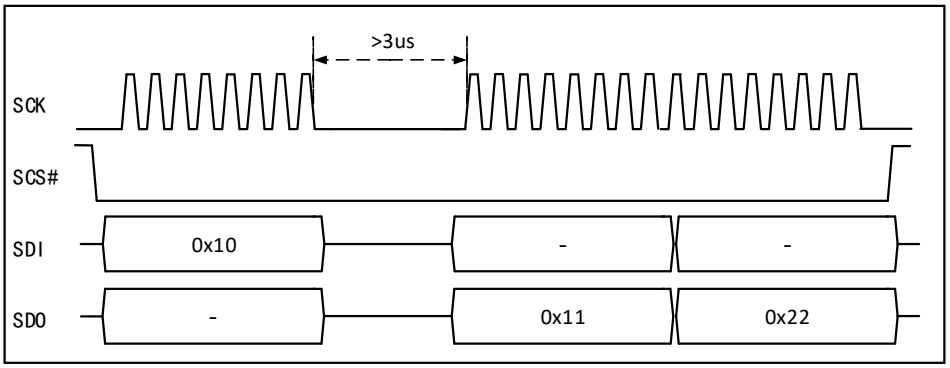
1). Set up FIFO controller



2). Read FIFO counter



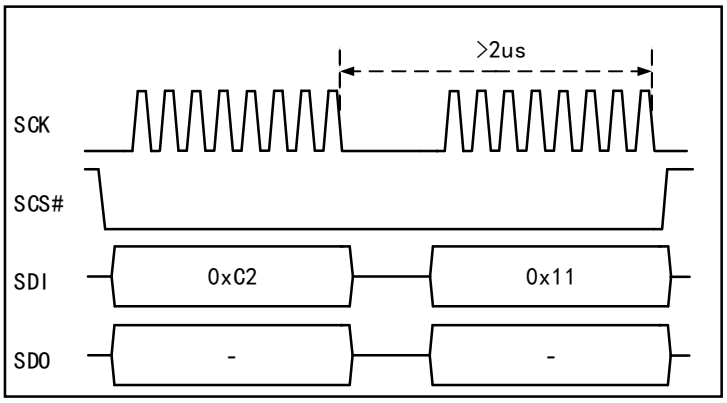
3). Read data



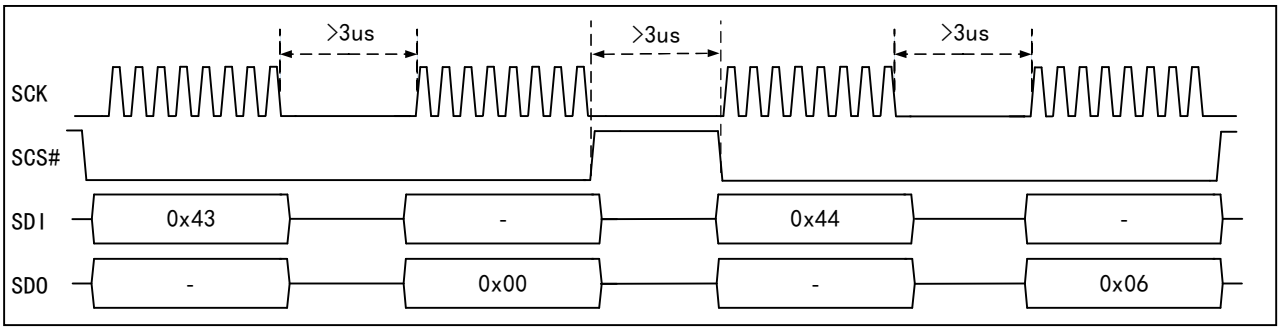
11.4 Write Serial Port Data

Example: CH9432 sends two bytes of data (0xAA and 0xBB) via serial port 1

1). Set up the FIFO controller



2). Read FIFO counter



3). Write data