CH346 Application Development Manual

V1.0

Catalog

1. Introduction

CH346 is a USB2.0 high speed converter chip to realize USB-UART, USB-SPI, USB-FIFO, and USB-GPIO interfaces. CH346C supports 3 working modes, which need to be selected according to the application.

CH346DLL is used to provide the CH346 chip with the parallel FIFO and SPI interface operation functions on the operating system side, and supports the vendor's driver interface, so there is no need to distinguish between the driver interface and the chip's operating mode when using it.

2. Interface Description

According to the characteristics of USB converter interface supported by CH346, CH346DLL provides interface functional functions for parallel FIFO and SPI, including the basic functional function and the corresponding functional function, For example, for FIFO/SPI read/write, when setting the internal buffered upload mode, the read speed is higher than the direct upload mode; and also when setting the internal buffered download mode, the write speed is higher than the direct download mode.

The interfaces supported by the CH346C are shown in the table below. Different operating modes can be switched by configuring the pin level status via MODE at power-up or by configuring the EEPROM.

Working Mode	Functional Interface Description	Driver Interface	API
Mode 0	Interface 0:USB2.0 to High-speed UART0	CH343SER(VCP)	Native UART API in the system
	Interface 1: USB2.0 to Passive Parallel FIFO	CH346DRV	CH346xxx in CH346DLL
Mode 1	Interface 0:USB2.0 to High-speed UART0	CH343SER(VCP)	Native UART API in the system
	Interface 1:USB2.0 to Passive SPI interface	CH346DRV	CH346xxx in CH346DLL
Mode 2	Interface 0:USB2.0 to High-speed UART0	CH343SER(VCP)	Native UART API in the system
	Interface 1:USB2.0 to High-speed UART1		

Table. CH346 Interface function API

3. FIFO/SPI interface

3.1 Related data types

3.1.1 Device Information

```
typedef struct _DEV_INFOR{
        UCHAR           iIndex;                     // Currently open serial number
        UCHAR           DevicePath[MAX_PATH];       // Device link name, used in CreateFile
        UCHAR           UsbClass;                   // Driver category
        UCHAR           FuncType;                   // Functional category
        CHAR           DeviceID[64];                // USB\VID_xxxx&PID_xxxx
```

```
    UCHAR          ChipMode;                // Chip working mode
                                            0: Mode 0(UART0+FIFO),
                                            1: Mode 1(Uart0+SPI),
                                            2: Mode 2(Uart0/1),
    HANDLE         DevHandle;               // The device handle
    USHORT         BulkOutEndpMaxSize;      // Bulk upload endpoint size
    USHORT         BulkInEndpMaxSize;       // Bulk download endpoint size
    UCHAR          UsbSpeedType;            // USB speed type, 0: FS, 1: HS, 2: SS
    UCHAR          CH346IfNum;              // USB interface number
    UCHAR          DataUpEndp;              // Bulk upload endpoint address
    UCHAR          DataDnEndp;              // Bulk download endpoint address
    CHAR           ProductString[64];       // USB product string
    CHAR           ManufacturerString[64];  // USB vendor string
    ULONG          WriteTimeout;            // USB write timeout
    ULONG          ReadTimeout;             // USB read timeout
    CHAR           FuncDescStr[64];         // Interface functional descriptor
    UCHAR          FirewareVer;             // Firmware version, hexadecimal value
}mDeviceInforS,*mPDeviceInforS;
```

3.2 Public operation functions

3.2.1 CH346OpenDevice

Function description

This function is used to open the CH346 device, supports opening in CH346 FIFO/SPI interface mode.

Function definitions

```
    HANDLE    WINAPI
    CH346OpenDevice(ULONG    iIndex);
```

Parameter description

iIndex: Specifies the device number

Return value

Returns the device serial number if the execution is successful.

3.2.2 CH346CloseDevice

Function description

This function is used to close the CH346 device, supports closing in CH346 FIFO/SPI interface mode.

Function definitions

```
    VOID    WINAPI
    CH346CloseDevice(ULONG    iIndex)
```

Parameter description

iIndex: Specifies the device number

3.2.3 CH346SetDeviceNotify

Function description

This function is used to specify the device event notification function, it can be used for dynamic hot plug detection of FIFO/SPIinterfaces mode for CH346

Function definitions

```
BOOL    WINAPI
CH346SetDeviceNotify(ULONG          iIndex,
                     PCHAR          iDeviceID,
                     mPCH346_NOTIFY_ROUTINE      iNotifyRoutine)
```

Parameter description

iIndex: Specifies the device number, number 0 corresponds to the first device

iDeviceID: Optional parameter, pointing to a string, specifies the ID of the monitored device, the string terminated with \0.

iNotifyRoutine: Specify the device event callback routine. If it is NULL, event notification is cancelled. Otherwise the routine is called when the event is detected.

Return value

The return value is TRUE on success and FALSE on failure

3.2.4 CH346GetDeviceInfor

Function description

This function is used to get the current interface mode and VID/PID of the device.

Function definitions

```
BOOL    WINAPI
CH346GetDeviceInfor(ULONG          iIndex,
                    mDeviceInforS  *DevInformation)
```

Parameter description

iIndex: Specify the serial number of the operating device

DevInformation: Device information structure

Return value

The return value is TRUE on success and FALSE on failure

Annotations

Device information structure, see DEV_INFOR

3.2.5 CH346GetSnString

Function description

This function is used to get the USB serial number string

Function definitions

```
BOOL    WINAPI
CH346GetSnString (ULONG       iIndex,
                    PVOID         oBuffer,
                    PUCHAR        iSerialNumberStr)
```

Parameter description

iIndex: Specifies the device number

oBuffer: Pointer to a buffer large enough to store the descriptor.

iSerialNumberStr: Pointer to the serial number of the acquired device

Return value

The return value is 1 on success and 0 on failure

3.2.6 CH346ResetDevice

Function description

This function is used to reset the USB device.

Function definitions

```
BOOL    WINAPI
CH346ResetDevice (ULONG       iIndex)
```

Parameter description

iIndex: Specifies the device number

Return value

The return value is TRUE on success and FALSE on failure

3.2.7 CH346SetTimeout

Function description

This function is used to set timeout for USB data reading and writing.

Function definitions

```
BOOL     WINAPI
CH346SetTimeout(ULONG       iIndex,
                ULONG       iWriteTimeout,
                ULONG       iReadTimeout)
```

Parameter description

iIndex: Specifies the device number

iWriteTimeout: Specify timeout for USB write-out data blocks, the unit is millisecond (mS), 0xFFFFFFFF specifies no timeout (default)

iReadTimeout: Specify the timeout for USB read data blocks, the unit is millisecond (mS), 0xFFFFFFFF specifies no timeout (default)

Return value

The return value is TRUE on success and FALSE on failure

3.2.8 CH346GetDeviceName

Function description

This function is used to get the device name.

Function definitions

```
PVOID WINAPI
CH346GetDeviceName (ULONG      iIndex)
```

Parameter description

iIndex: Specifies the device number

Return value

The return value is TRUE on success and FALSE on failure

3.2.9 Interface dynamic hot plug detection

Detection of synchronous serial interface dynamic hot plug information can be achieved through the CH346SetDeviceNotify function, the code reference is as follows.

Enable the monitoring of USB plug and unplug of CH346 synchronous serial port:

```
CH346SetDeviceNotify(DevIndex, USBDevID, UsbDevPnpNotify);
```

Disable the monitoring of USB plug and unplug of CH346 synchronous serial port, be sure to close the program when it exits.

```
CH346SetDeviceNotify(DevIndex, USBDevID, NULL);
```

```
// CH346 device hot plug detection notification program

#define      CH346_DEVICE_ARRIVAL    3          // Device plug event, already plugged

#define      CH346_DEVICE_REMOVE_PEND   1     // Device will be unplugged

#define      CH346_DEVICE_REMOVE       0      // Device unplug event, already unplugged
VOID     CALLBACK     UsbDevPnpNotify (ULONG iEventStatus )

{

    if(iEventStatus==CH346_DEVICE_ARRIVAL)         // Device plug event, already plugged
        PostMessage(DebugHwnd,WM_CH346DevArrive,0,0);

    else if(iEventStatus==CH346_DEVICE_REMOVE)     // Device unplug event, already unplugged
        PostMessage(DebugHwnd,WM_CH346DevRemove,0,0);

    return;

}
```

3.2.10 Device enumeration operation

In this library, the API implements corresponding operations by specifying device serial number. The device serial number is generated based on the sequence of devices being inserted one by one. The device enumeration function can be realized by opening the corresponding device serial number through the device Open function, and judging whether the device is valid and exists according to the function return value.

The FIFO/SPI interface open/close functions are available: CH346OpenDevice/CH346CloseDevice.

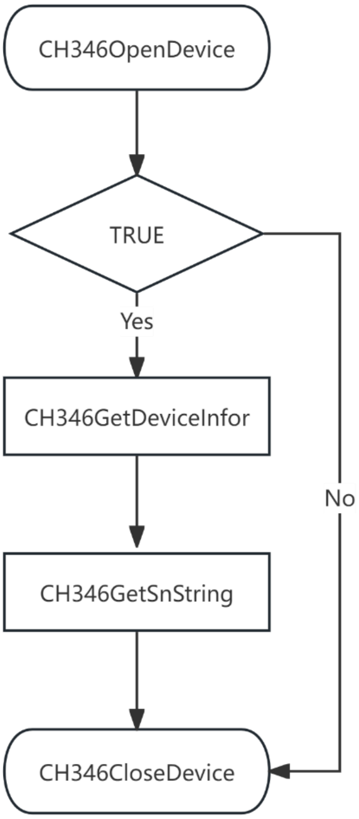

Figure 3.2.10 Device enumeration flowchart

3.3 FIFO/SPI Interface Functions

3.3.1 Operation Process

After the device is enabled, set the device USB read and write timeout parameters, then configure the interface mode (mode 0: parallel FIFO, mode 1: SPI interface)， After successful setup, you can communicate with the device by calling the FIFO/SPI read/write function（Enable internal buffered upload mode and internal buffered download mode to increase the FIFO/SPI interface read/write speed）.

The function call flowchart is as follows:

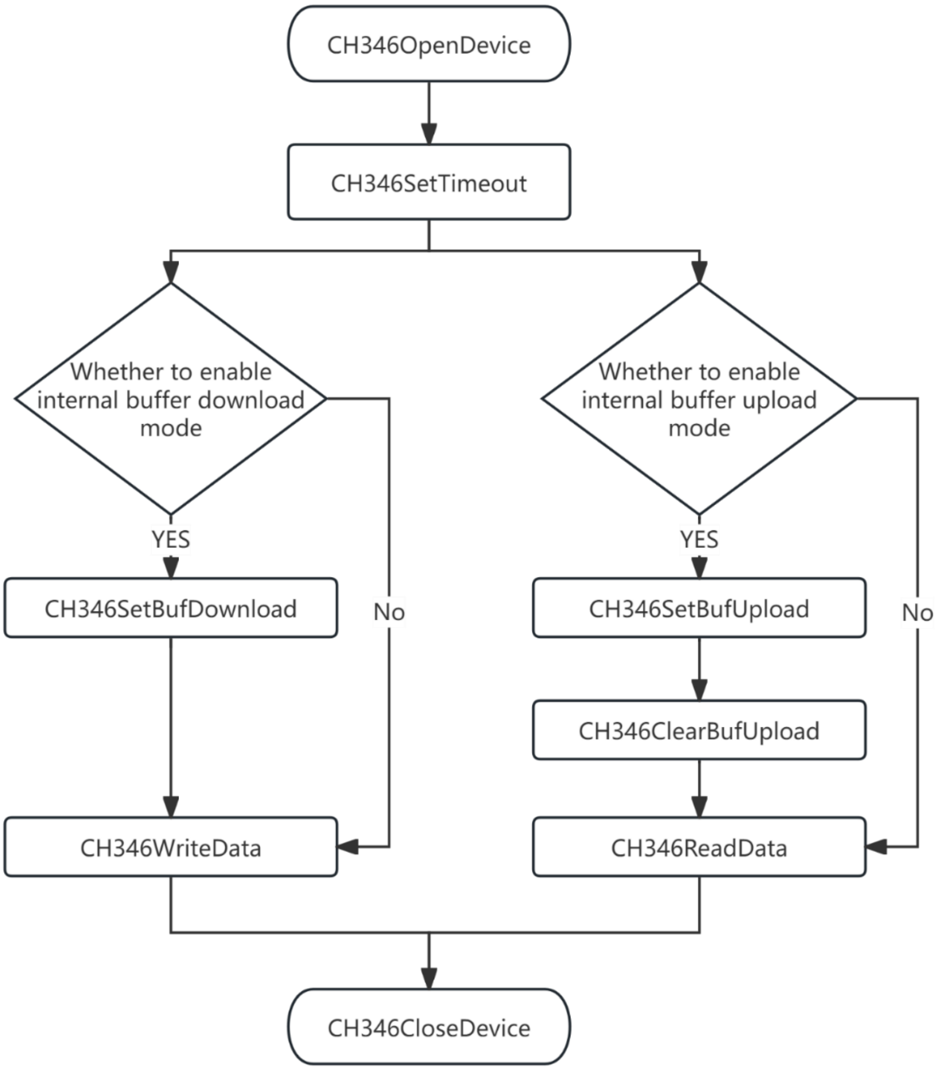

Figure 3.3.1 FIFO/SPI function operation flowchart

For details about the function, see the following.

3.3.2 CH346SetChipMode

Function description

This function is used to set the working mode of the chip.

Function definitions

```
BOOL    WINAPI
CH346SetChipMode (ULONG           iIndex,
                  UCHAR         iChipMode,
                  BOOL          iIsSave)
```

Parameter description

iIndex:	Specifies the device number
iChipMode:	Chip working mode,0: Uart + FIFO; 1:Uart + SPI
iIsSave:	Enable power-off configuration

Return value

The return value is TRUE on success and FALSE on failure

3.3.3 CH346ReadData

Function description

This function is used for data reading in FIFO/SPI interface mode. In mode 0, it gets the data received from the hardware parallel FIFO through USB; in mode 1, it gets the data received from the hardware SPI interface through USB.

Function definitions

```
BOOL    WINAPI
CH346ReadData (  ULONG        iIndex,
                 PVOID        oBuffer,
                 PULONG       ioLength)
```

Parameter description

iIndex: Specifies the device number

oBuffer: Pointer to a buffer large enough to store the read data, not exceeding 4MB.

ioLength: Pointer to the length unit; when entered, it is the length to be read, and when returned, it is the actual length to be read.

Return value

The return value is TRUE on success and FALSE on failure

3.3.4 CH346WriteData

Function description

This function is used to write data in FIFO/SPI interface mode. In mode 0, it sends data to parallel FIFO through USB; in mode 1, it sends data to SPI through USB.

Function definitions

```
BOOL    WINAPI
CH346WriteData (ULONG      iIndex,
                PVOID      oBuffer,
                PULONG    ioLength)
```

Parameter description

iIndex: Specifies the device number

oBuffer: Pointer to a buffer,to store the data ready to be write-out.

ioLength: Pointer to the length unit; when entered, it is the length to be read, and when returned, it is the actual length to be read.

Return value

The return value is TRUE on success and FALSE on failure

3.3.5 CH346AbortRead

Function description

This function is used to abort the current USB read operation, applicable to direct USB read operations (non-buffered upload mode).

Function definitions

```
BOOL    WINAPI
CH346AbortRead (ULONG        iIndex)
```

Parameter description

iIndex: Specifies the device number

Return value

The return value is TRUE on success and FALSE on failure

3.3.6 CH346AbortWrite

Function description

This function is used to abort the current USB write operation, applicable to direct USB write operation (non-buffered upload mode).

Function definitions

```
BOOL    WINAPI
CH346AbortWrite (ULONG        iIndex)
```

Parameter description

iIndex: Specifies the device number

Return value

The return value is TRUE on success and FALSE on failure

3.3.7 CH346SetBufUpload

Function description

Enable internal buffered upload mode,Read speed is higher than the direct upload mode, the size of the uploaded packet must be the size set by BufSize; If enabled, the driver actively polls for data via USB, which is acquired by the hardware parallel FIFO or SPI interface, and caches the data in the driver buffer. The CH346ReadData function retrieves data directly from the internal buffer.

Function definitions

```
BOOL WINAPI
CH346SetBufUpload (ULONG     iIndex,
                   ULONG     iEnableOrClear,
                   ULONG     BufSize);
```

Parameter description

iIndex: Specifies the device number, number 0 corresponds to the first device

iEnableOrClear: 0: disable internal buffered upload mode, use direct upload;
Non-zero: enable internal buffered upload mode and clear the existing data in the buffer.

BufSize: Buffer size per packet, maximum 4MB.

Return value

The return value is TRUE on success and FALSE on failure

3.3.8 CH346QueryBufUpload

Function description

This function is used to query the number of packets and total bytes of data already in the internal upload buffer.

Function definitions

```
BOOL WINAPI
CH346QueryBufUpload ( ULONG      iIndex,
                      PULONG     oPacketNum,
                      PULONG     oTotalLen);
```

Parameter description

iIndex: Specifies the device number, number 0 corresponds to the first device

oPacketNum: Returns the number of received packets in the internal buffer

oTotalLen: Returns the total number of bytes of data received in the internal buffer, which is a multiple of the packet size BufSize.

Return value

The return value is TRUE on success and FALSE on failure

3.3.9 CH346ClearBufUpload

Function description

This function is used to flush the internal upload buffer without pausing the internal buffer upload mode.

Function definitions

```
    BOOL    WINAPI
    CH346ClearBufUpload (ULONG          iIndex);
```

Parameter description

iIndex: Specifies the device number, number 0 corresponds to the first device

Return value

The return value is TRUE on success and FALSE on failure

3.3.10 CH346SetBufDownload

Function description

This function is used to set the internal buffered download mode, the write speed is higher than the direct download mode. If enabled, when CH346WriteData writes data, the data will be cached firstly in the driver's internal download buffer and then returned directly. The driver polls the data via USB and sends it to the hardware parallel FIFO or SPI interface.

Function definitions

```
    BOOL    WINAPI
    CH346SetBufDownload ( ULONG iIndex,
                          ULONG       iEnableOrClear,
                          ULONG       iPacketCnt );
```

Parameter description

iIndex: Specifies the device number, number 0 corresponds to the first device

iEnableOrClear: 0: disable internal buffered download mode, use direct download; non-0: enable internal buffered download mode and clear the existing data in the buffer.

iPacketCnt: Maximum number of packets to cache, up to 10 packets

Return value

The return value is TRUE on success and FALSE on failure

3.3.11 CH346QueryBufDownload

Function description

This function is used to query the number of packets remaining in the internal download buffer (not yet sent), and returns the number of packets if successful.

Function definitions

```
BOOL    WINAPI
CH346QueryBufDownload (ULONG   iIndex,
                       ULONG           oPacketCnt,
                       ULONG           oTotalLen);
```

Parameter description

iIndex: Specifies the device number, number 0 corresponds to the first device

oPacketCnt: Returns the number of packets remaining in the internal buffer

oTotalLen: Returns the total number of remaining bytes in the internal buffer

Return value

The return value is TRUE on success and FALSE on failure