CC3100 Host Interface

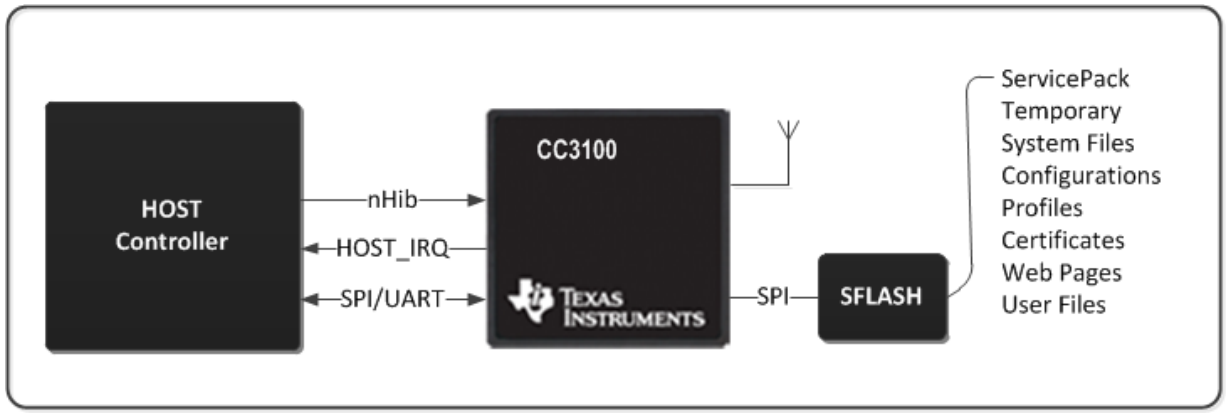
Return to CC31xx & CC32xx Home Page

Contents

Introduction

The SimpleLink CC3100 is a Wi-Fi and networking device provides a comprehensive networking solution for low-cost and low-power micro-controllers using a thin driver and simple APIs set. The host thin driver is a multi-platform Ansi-C driver (C89) compatible for different types of 8/16/32 micro-controllers, big or little endian and running with or without operating system. The device interfaces to an external host controller using standard SPI or UART physical interfaces and provides additional auxiliary line (HOST_IRQ) to allow better and simpler power management of the host controller.

The following figure shows the application diagram of the SimpleLink CC3100 Wi-Fi and networking solution:



SimpleLink CC3100 Wi-Fi and Networking solution

The porting of the SimpleLink host driver to a new platform is based on few simple steps. One of the major and important steps is writing the host communication interface layer. The implementation of this layer affects directly the efficiency and stability of the SimpleLink host driver. The purpose of this document is to provide the complementary information on the SimpleLink™ CC3100 host interfaces required for implementing this layer correctly. It covers the main attributes of the host interface protocol including supported modes, structures of different commands and communication flow. The document also provides guidelines for selecting the configuration of the host side. It should be used by system and software engineers during early integration stages.

The host communication interface layer consists of 5 interface functions that should be provided to the SimpleLink host driver:

Function Name	Description
sl_IfOpen	Opens the interface communication port to be used for communicating with the SimpleLink device
sl_IfClose	Closes an opened interface communication port
sl_IfWrite	Transmits len bytes of data to the interface communication channel according to opened attributes
sl_IfRead	Attempts to read up to len bytes from an opened communication channel into a buffer
sl_IfRegIntHdlr	Register an interrupt handler routine for the Host IRQ

Terminology and Abbreviation

Abbreviation	Meaning
Host	Host refer to an embedded controller running the SimpleLink driver and use the SimpleLink device as a networking peripheral
UART	Universal asynchronous receiver/transmitter used for serial communication between the Host the the SimpleLink device
SPI	Serial Peripheral Interface a synchronous serial data link between the Host and the SimpleLink device
MISO	Master In Slave Out SPI line
MOSI	Master Out Slave In SPI line
SYNC	Synchronization word
RTS	Request to send. In this document refer to UART hardware flow control line
CTS	Clear to send. In this document refer to UART hardware flow control line

Host communication protocol and flows

Message Types

The SimpleLink Host protocol consists of 4 message types:

Command	Any message from the Host to the SimpleLink device that is not data message
Command Complete	Replay message from the SimpleLink device to the Host. Sent as a reply to any command
Data	Special message from the Host to the SimpleLink device containing data to be transmitted over the air
Async Event	Asynchronous message from the SimpleLink device to the Host

Note: The SimpleLink device support handling of a single command at a given time. Command handling is completed when the device is sending “Command Complete” message.

Message Format

Host to SimpleLink Device			
SYNC	**OP+LEN**	**DESCRIPTORS**	**PAYLOAD**
32bit	32bit (16bit+16bit)	Changed per OP (Aligned to 32bit)	Variable (Aligned to 32bit)

SimpleLink Device to Host				
SYNC	**OP+LEN**	**STATUS**	**DESCRIPTORS**	**PAYLOAD**
32bit	32bit (16bit+16bit)	32bit	Changed per OP (Aligned to 32bit)	Variable (Aligned to 32bit)

Description:

SYNC	Synchronization word, used by the protocol. For more details see below.
OP+LEN	Identifier of the message consist of two fields: ▪ Opcode 16bit Unique number identify the message ▪ Length The length of the message including the STATUS, DESCRIPTORS and PAYLOAD fields
STATUS	32bit bit-field value determines the status of the device
DESCRIPTORS	The set of parameters attached for a opcode. The size of this field is constant for each opcode. For some message the length of this field could be 0
PAYLOAD	This field contains the variable length data of a message

Synchronization words

The SimpleLink host protocol uses synchronization words to keep the host and the device in sync by allowing the entities to find a beginning of a message. There are three types of synchronization words in use:

- Host to Device – write
- Host to Device – read (relevant for SPI only)
- Device to host

Each synchronization word is 4 bytes long and consists mostly of a constant pattern. The patterns are given in the following table:

Sync Word	Pattern
Host to Device, Write	0x4321123x (x = 4b'01xx)
Host to Device, Read	0x8765567x (x = 4b'11xx)
Device to Host	0xABCDDCBx (x = 4b'11xx)

The two LSBits of the sync word can have any value managed by the driver and the device.

Notice the SYNC words might appear differently on the physical lines, depending on the endianity and word size of the host.

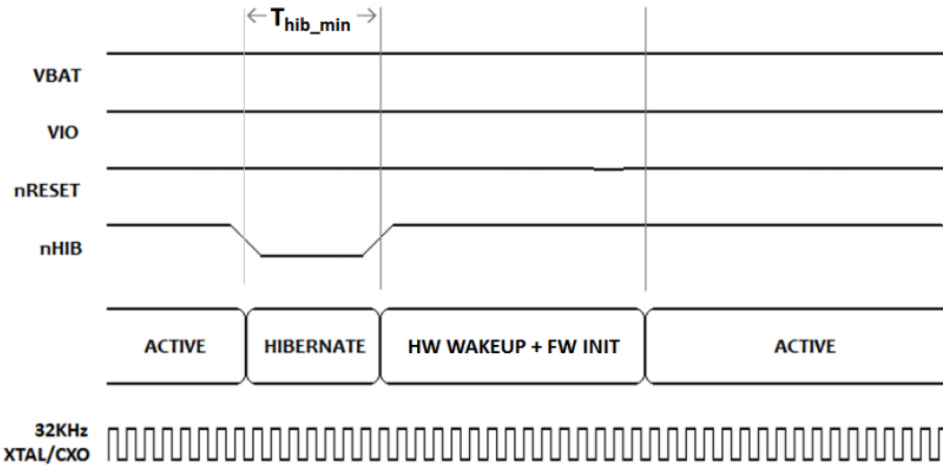
The Host IRQ Line

An auxiliary interrupt line from the CC3100 device to the host MCU is used to inform the host when the SimpleLink device has message to send to the Host. This line is active high, and can be treated as either edge or level (per host implementation).

Since the interrupt line is tri-stated while the SimpleLink device is disabled, it is important that the host environment keeps it pulled down at those times, either by an external pull-down resistor, or by an internal pull-down resistor in the Host MCU's I/O. Having the interrupt line pulled down at logical '0' during initialization is crucial, since otherwise a false interrupt might be detected by the host prior to the initialization complete state of the SimpleLink device.

When implementing the host IRQ using level and not edge it is important to implement additional two functions: sl_IfMaskIntHdlr and sl_IfUnMaskIntHdlr. These functions would be called by the driver to enable/disable the interrupt.

nHIB timing requirements



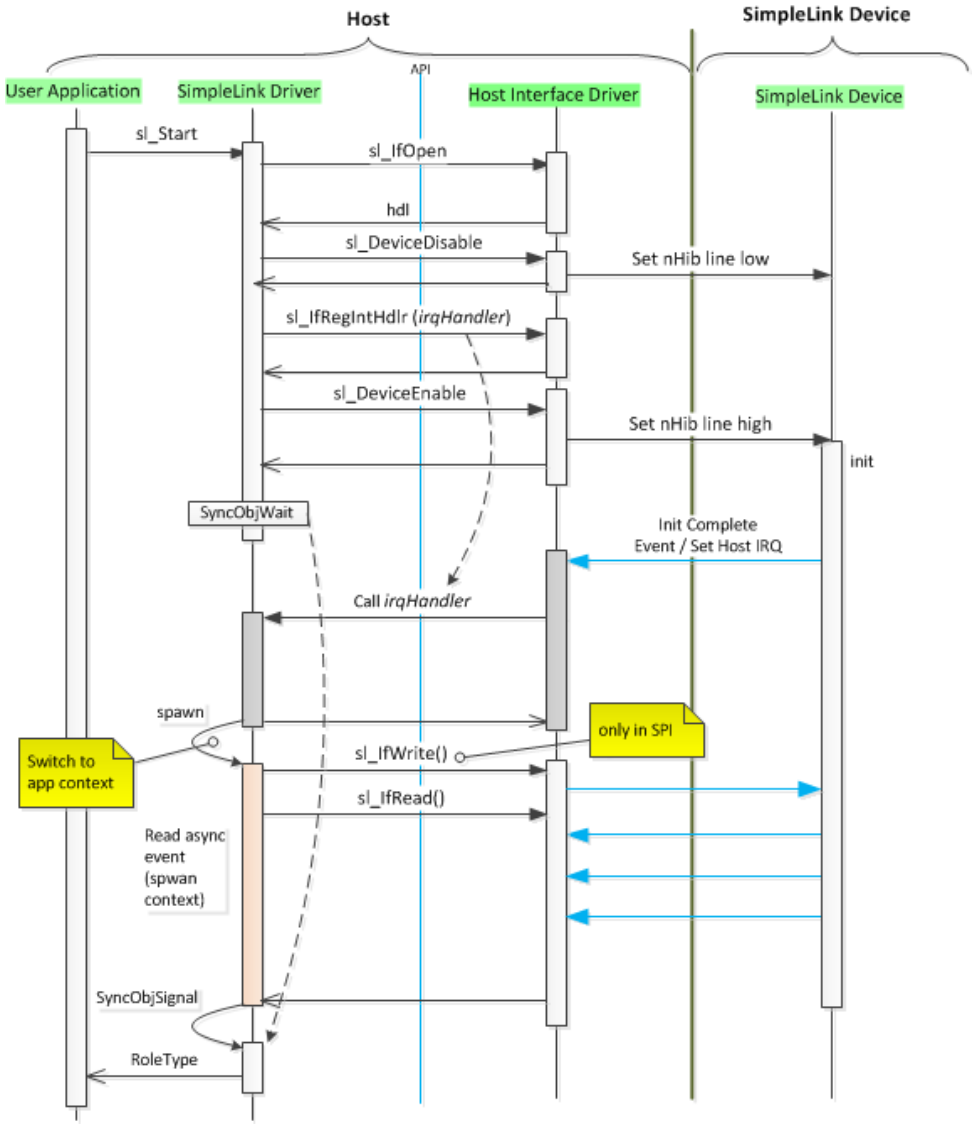
CC3100 nHIB timing requirement

Minimum Hibernate time required	**10 ms**

Initialization flow

During initialization, nHIB pin is asserted (to enable the device), while the nRESET pin is kept high. At this stage, HOST_IRQ pin should be driven low by the device until initialization is complete. During this time, and until HOST_IRQ is asserted for the first time, the host must not communicate with the device otherwise the communication with the device might not be established.

In Host UART topologies that the HOST_IRQ line is not in use, the first received byte indicates that the communication with the device could be established.



Basic Initialization Flow

SPI Interface

Main article: CC3100 SPI Host Interface

SPI is a de-facto industry standard and many different configurations for it exists.

UART Interface

Main article: CC3100 UART Host Interface

UART is a standard asynchronous serial communication that works between two entities and have a support for hardware flow control. In UART interface there is no Master/Slave relationship defined by the Hardware and each entity can send data to the other side independently in full duplex mode. The hardware flow control makes use of two hardware lines, RTS (Request to Send) and CTS (Clear to Send) to allow each side indicate to the other side if is ready to handle data.

Links

{{#invoke: Navbox | navbox }}

{{

1. switchcategory:MultiCore=
   - For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum
   - For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum

   Please post only comments related to the article **CC3100 Host Interface** here.

Keystone=
- For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum
- For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum

Please post only comments related to the article **CC3100 Host Interface** here.

C2000=*For technical support on the C2000 please post your questions on The C2000 Forum. Please post only comments about the article **CC3100 Host Interface** here.*

DaVinci=*For technical support on DaVincoplease post your questions on The DaVinci Forum. Please post only comments about the article **CC3100 Host Interface** here.*

MSP430=*For technical support on MSP430 please post your questions on The MSP430 Forum. Please post only comments about the article **CC3100 Host Interface** here.*

OMAP35x=*For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article **CC3100 Host Interface** here.*

OMAPL1=*For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article **CC3100 Host Interface** here.*

MAVRK=*For technical support on MAVRK please post your questions on The MAVRK Toolbox Forum. Please post only comments about the article **CC3100 Host Interface** here.*

*For technical support please post your questions at http://e2e.ti.com. Please post only comments about the article **CC3100 Host Interface** here.*

}}

Links

Amplifiers & Linear
Audio
Broadband RF/IF & Digital Radio
Clocks & Timers
Data Converters
DLP & MEMS
High-Reliability
Interface
Logic
Power Management
Processors
- ARM Processors
- Digital Signal Processors (DSP)
- Microcontrollers (MCU)
- OMAP Applications Processors

Switches & Multiplexers
Temperature Sensors & Control ICs
Wireless Connectivity