

# CC32xx Deepsleep Application

---

## Overview

DeepSleep is one of the low power modes supported by the CC3200 device. There are various options available to select the module to be active during a low power mode by appropriately configuring the designated registers of the PRCM module and thereby causing an exit from the low power mode based on activity on the chosen active module. The clock to the other modules, that are not chosen to be active, get disabled. The clock is also halved in case of DeepSleep mode.

The various modules that can be clock gated (enabled/disabled) are:

1. UART
2. UDMA
3. TIMER
4. WDT
5. GPIO
6. I2C

Also in the deep sleep mode, activity on the WiFi network triggered by the NWP can also be chosen as a cause for the wakeup.

## Application details

The objective of this application is to showcase the deepsleep as a power saving tool in a networking context (in this case a TCP server). In this example the Idle task hook function available to the application as a FreeRTOS feature is used to enter the deepsleep power mode. Whenever there is activity on the network, the device is brought out of deepsleep state and services the networking sequence. So whenever the networking APIs block, the device enters deepsleep on idle task execution.

## Source Files briefly explained

- **main** - Core TCP server implementation. Idle task hook function implementation that enters deepsleep is also present.
- **nw\_if** - Common functions to handle connection to AP and FreeRTOS hook functions.
- **gpio\_if** - Basic GPIO interface APIs.
- **pinmux** - Assigns a personality to the pins at the device boundary.
- **uart\_if** - To display status information over the UART
- **timer\_if** - Wrapper for timer module driver
- **startup\_\*** - Tool specific vector table implementation.

[Return to CC31xx & CC32xx Home Page](#)[Return to CC31xx Sample Applications](#)

## Usage

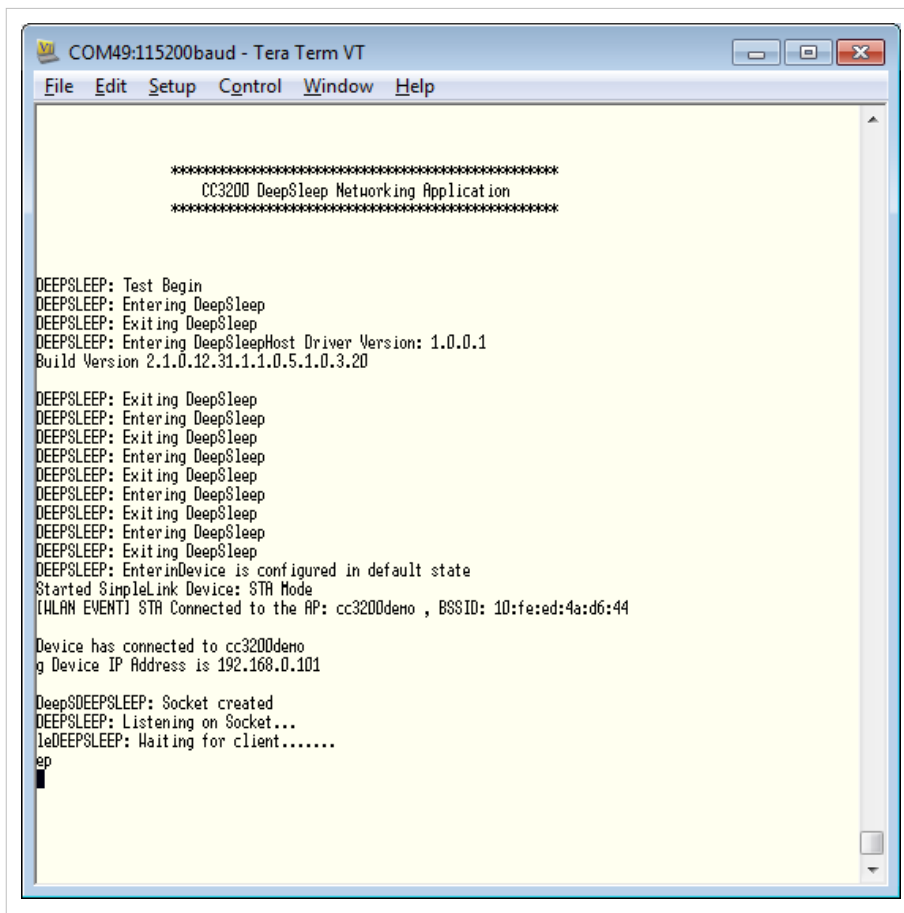
**Note:** It is recommended that the application be tested by flashing the .bin file onto the serial flash on LaunchPad only. The IAR and CCS debuggers will disconnect on entering low power modes and cannot be used for subsequent debug.

- Setup a serial communication application (HyperTerminal/TeraTerm). For detail info visit CC31xx & CC32xx Terminal Setting.

On the host PC. The settings are:

- **Port:** Enumerated COM port
- **Baud rate:** 115200
- **Data:** 8 bit
- **Parity:** None
- **Stop:** 1 bit
- **Flow control:** None
- Run the reference application (Flashing the bin).
- Observe the status messages on the host over serial port to understand the sequence of operations performed by the application.
- The application tries to connect to AP by name DemoAP by default. In case this fails, the user is prompted to enter the AP name (Security type OPEN) to connect to on HyperTerminal/TeraTerm.
- Whenever the server starts waiting for a client, connect to the device from an external networking application. The TCP server port number is 5001 and the IP address is available over the UART logs.
- Notice the exit from the deepsleep whenever a new client connects. There are also other instances of the device entering deepsleep.

Terminal snapshot when application runs on device:



```
*****
CC3200 DeepSleep Networking Application
*****

DEEPSLEEP: Test Begin
DEEPSLEEP: Entering DeepSleep
DEEPSLEEP: Exiting DeepSleep
DEEPSLEEP: Entering DeepSleepHost Driver Version: 1.0.0.1
Build Version 2.1.0.12.31.1.1.0.5.1.0.3.20
DEEPSLEEP: Exiting DeepSleep
DEEPSLEEP: Entering DeepSleep
DEEPSLEEP: Exiting DeepSleep
DEEPSLEEP: Entering DeepSleep
DEEPSLEEP: Exiting DeepSleep
DEEPSLEEP: Entering DeepSleep
DEEPSLEEP: Exiting DeepSleep
DEEPSLEEP: Entering DeepSleep
DEEPSLEEP: Exiting DeepSleep
DEEPSLEEP: EnterinDevice is configured in default state
Started SimpleLink Device: STA Mode
[WLAN EVENT] STA Connected to the AP: cc3200demo , BSSID: 10:fe:ed:4a:d6:44

Device has connected to cc3200demo
g Device IP Address is 192.168.0.101

DeepSDEEPSLEEP: Socket created
DEEPSLEEP: Listening on Socket...
leDEEPSLEEP: Waiting for client.....
ep
```

## Limitations/Known Issues

The uart print to enter deepsleep is placed in the idle hook function which is of lowest priority and hence the print appear jumbled due to context switches. The print is retained only for clarity of the timing of entering deepsleep.

# Article Sources and Contributors

**CC32xx Deepsleep Application** *Source:* <http://processors.wiki.ti.com/index.php?oldid=184824> *Contributors:* A0221015, Codycooke, Jitgupta, Malokyle

# Image Sources, Licenses and Contributors

**File:Cc31xx cc32xx return home.png** *Source:* [http://processors.wiki.ti.com/index.php?title=File:Cc31xx\\_cc32xx\\_return\\_home.png](http://processors.wiki.ti.com/index.php?title=File:Cc31xx_cc32xx_return_home.png) *License:* unknown *Contributors:* A0221015

**File:Cc32xx return sample apps.png** *Source:* [http://processors.wiki.ti.com/index.php?title=File:Cc32xx\\_return\\_sample\\_apps.png](http://processors.wiki.ti.com/index.php?title=File:Cc32xx_return_sample_apps.png) *License:* unknown *Contributors:* A0221015

**Image:CC32xx Deep Sleep NW Terminal runScreen 1.0.0.png** *Source:* [http://processors.wiki.ti.com/index.php?title=File:CC32xx\\_Deep\\_Sleep\\_NW\\_Terminal\\_runScreen\\_1.0.0.png](http://processors.wiki.ti.com/index.php?title=File:CC32xx_Deep_Sleep_NW_Terminal_runScreen_1.0.0.png) *License:* unknown *Contributors:* Jitgupta