

CC32xx Websock Camera Application

Objective

This application demonstrates the real time JPEG streaming using websocket. The application captures VGA frames from camera sensor and send it over websocket to the web client such as browser.

Prerequisites

Hardware

- CC3200 LaunchPad Rev 3.2 onwards
- CC3200 Camera Booster Pack Rev 3.0 onwards
- MT9D111 Camera Module

Software

- CC3200 WebSocket application
- Latest SDK Firmware Package

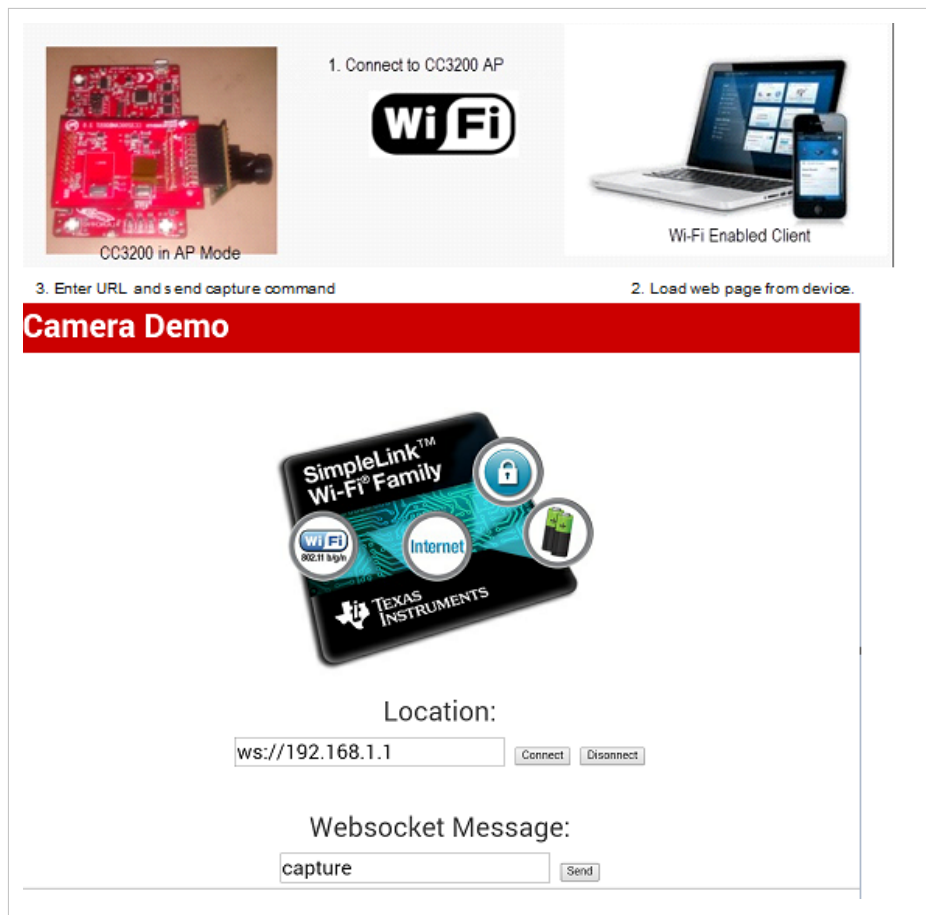
Application details

In this application, Simplelink device runs in Access Point Mode. User needs to connect the Wi-Fi enabled device PC/Smartphone to the device AP (Default SSID: mysimplelink-MACAddr)

Using a web browser, access http://mysimplelink.net/camera_demo.html, the static pages stored on Serial FLASH will be displayed.

The web page will ask for the URL of the CC3200 WebSocket Server. Default value is ws://192.168.1.1. Enter this and send 'Connect'. Wait for the alert window to confirm connection.

Send Capture command to device. The video streaming will now start. You can choose to send 'Disconnect' at any time. The streaming can be restarted by repeating the same process again. The following picture illustrates the usage of the camera application:



Websocket Camera Application Sequence

1. Client makes TCP Connection with Websocket Server at port 80/443.
2. The client initiates a websocket handshake by sending an upgrade request.
3. Websocket Server responds with Response code 101 if handshake is successful.
4. Client sees an alert on web page saying connection is established.
5. The client sends a capture command after which the device starts sending the JPEG frames as binary data without any polling from client.
6. Client closes connection by sending a close frame.

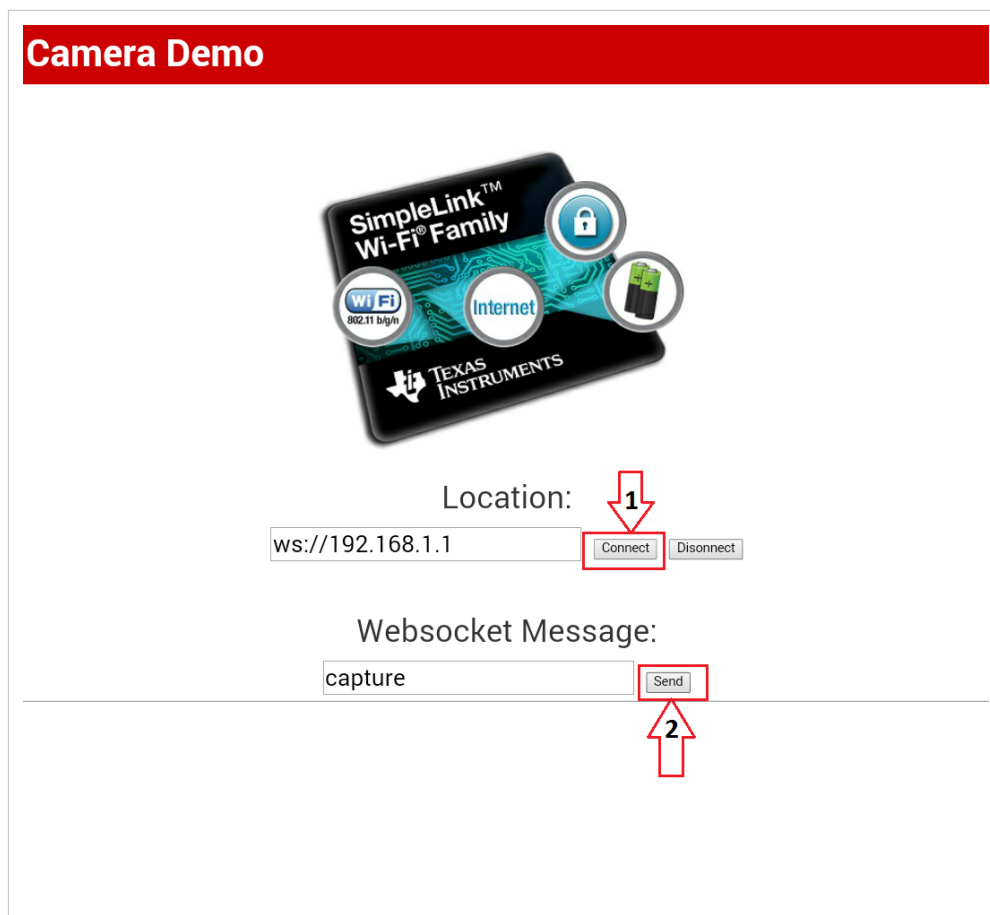
Source Files briefly explained

- main.c - System Initialization, Spawns the HTTPServer Task
- httpserverapp.c – Starts simplelink. Configures CC3200 in AP mode. Starts the Http server. Handles all websocket callbacks. Camera application is called from the callback.
- gpio_if.c - Handles GPIO related operations
- pinmux.c - Generated by Pinmux utility pin to configure Camera Data and I2C
- camera_app.c - Handles DMA Configuration, Camera Capture and NVMEM File operations
- network_if.c - Common network interface APIs
- udma_if.c - Common uDMA interface APIs
- MT9D111.c - MT9D111 sensor related functions are implemented
- startup_*.c - Implements tools specific interrupt vector table

WebSocket library is available under '<cc3200-sdk>\netapps\http\server' folder.

usage

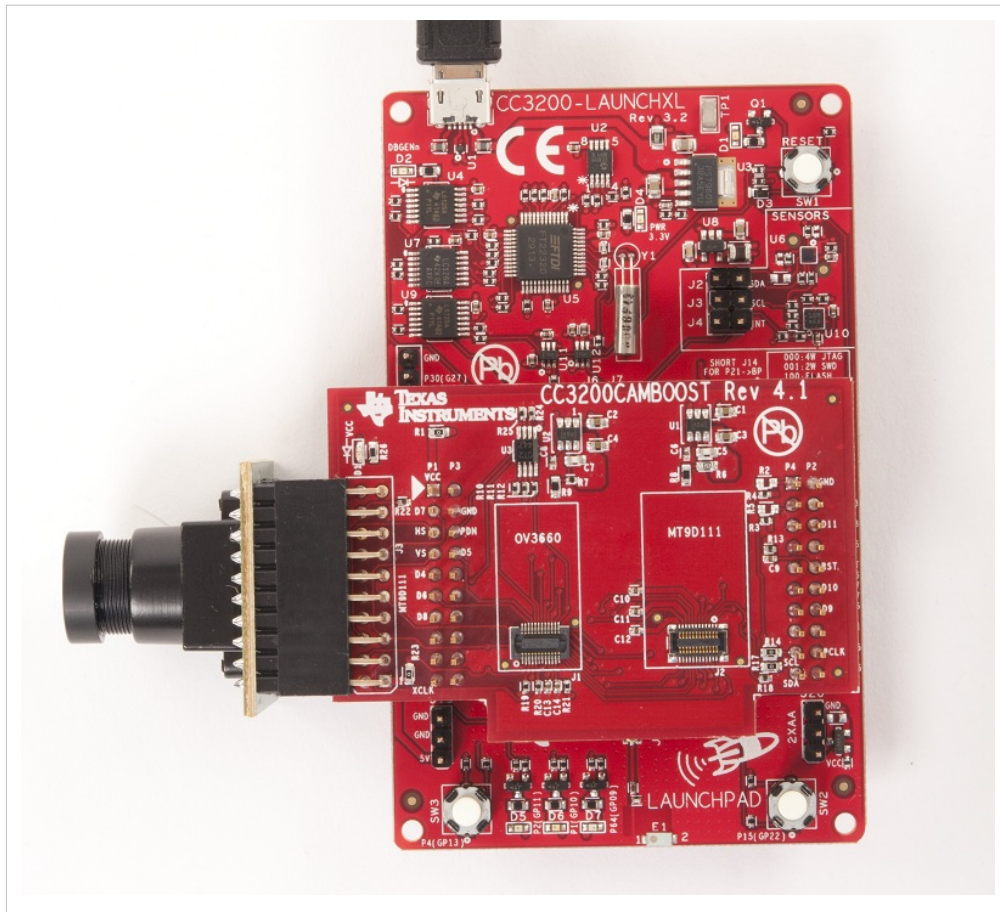
- Flash Example Content(html,image,Application binary)on serial flash by following below steps.
 - Open <cc3200-sdk>\examples\camera_application\html\ websocket_application.ucf session file in Uniflash.
 - Flash the files to the device. Detailed instructions at Uniflash User Guide.
- Configure the device as described in next section
- Run the reference application
 - Run From IDE (IAR/CCS)
 - Open the Project as mentioned in the <cc3200-sdk>\docs\CC3200-Getting Started Guide.pdf
 - Build and download the application to the board
 - Run Flashed Binary file
 - Remove SOP Jumpers on Board and Press Reset. Refer <cc3200-sdk>\docs\CC3200-Getting Started Guide.pdf for detailed Information.
- Device started in AP mode, Connect your PC/SmartPhone to 'mysimplelink-<macAddress>'.
- Open 'mysimplelink.net/camera_demo.html' in browser of PC/MAC/Smartphone, goto 'Camera' tab and Control Camera Capture in the CC3200 device.



- Click on Connect button and wait for a notification pop-up message saying 'WebSocket Connected'.
- After websocket connected click on send button to get the live stream, it may take 10-20 sec for stream to get started on first time.

Configuring the target device

1. Remove J10 and J11 jumpers to cut off the TDI and TDO from FTDI.
2. Mount the jumpers J6 and J7 in the RX/TX mode. This enables the launchpad interface to the booster pack.
3. Remove SOP-2 Jumper. For Debug(Using IAR/CCS) Mode, Ensure SOP 0 is mounted on the CC3200 launchpad. Since, TDI and TDO are used for I2C interface, the JTAG is operational in SWD mode.
4. Remove Jumpers J2, J3 and J4. This disconnects the GPIO from on-board sensors.
5. Mount the Camera Booster pack on the launchpad as shown below. Make sure the VCC on P1 connector mates with VCC indication on the booster pack. Similarly, the GND on P2 connector should align with GND indication on the camera booster pack.



Please note that only one sensor has to be interfaced to the camera booster pack.

Limitation/Known Issues

- The application requires about 250 KB space in serial Flash. Downloading html/bin files using uniflash may fail if enough space is not available on serial flash. Try erasing older files from serial flash using Uniflash.
- Maximum Frame rate is 5 fps which may vary with different camera sensor.

Article Sources and Contributors

CC32xx Websock Camera Application *Source:* <http://processors.wiki.ti.com/index.php?oldid=185961> *Contributors:* Jitgupta, Kaushal

Image Sources, Licenses and Contributors

Image:CC32xx_WebSock_Camera_webpage_1.png *Source:* http://processors.wiki.ti.com/index.php?title=File:CC32xx_WebSock_Camera_webpage_1.png *License:* unknown *Contributors:* Jitgupta

Image: CC32xx WebSock Camera Usage webpage.png *Source:* http://processors.wiki.ti.com/index.php?title=File:CC32xx_WebSock_Camera_Usage_webpage.png *License:* unknown *Contributors:* Jitgupta

Image:LP_Camera_mount.jpg *Source:* http://processors.wiki.ti.com/index.php?title=File:LP_Camera_mount.jpg *License:* unknown *Contributors:* Kaushal
