

Skywire[®] Development Kit

Socket Dial Example

NimbeLink Corp

Updated: October 2022



© NimbeLink Corp. 2022. All rights reserved.

NimbeLink Corp. provides this documentation in support of its products for the internal use of its current and prospective customers. The publication of this document does not create any other right or license in any party to use any content contained in or referred to in this document and any modification or redistribution of this document is not permitted.

While efforts are made to ensure accuracy, typographical and other errors may exist in this document. NimbeLink reserves the right to modify or discontinue its products and to modify this and any other product documentation at any time.

All NimbeLink products are sold subject to its published Terms and Conditions, subject to any separate terms agreed with its customers. No warranty of any type is extended by publication of this documentation, including, but not limited to, implied warranties of merchantability, fitness for a particular purpose and non-infringement.

NimbeLink is a registered trademark, and Skywire is a registered trademark, of NimbeLink Corp. All trademarks, service marks and similar designations referenced in this document are the property of their respective owners.

Table of Contents

Table of Contents	2
1. Introduction	3
1.1. Prerequisites	3
2. Socket Dial	4
2.1. Overview	4
2.2. Setup APN (If Applicable)	4
2.3. Configure Socket (If Applicable)	5
2.4. Activate Skywire Socket	5
2.5. Initiate Socket Dial	6
2.6. Send Data via HTTP	7
2.7. Receive Data via HTTP	7
2.8. Shutdown the Socket Connection	8
3. Working Example 1: dweet.io	8
3.1. Overview	8
3.2. Get IMEI of Modem	9
3.3. Initiate Socket Dial	9
3.4. Send Data via HTTP	9
3.5. Receive Data via HTTP	10
4. Troubleshooting	11
4.1 New Line/Carriage Return	11
5. Appendix – Socket Dial Flow Charts	11
5.1. Introduction	11
5.2. NL-SW-HSPA Flow Chart	12
5.3. NL-SW-LTE-TSVG-x Flow Chart	13
6. Document Version Information	14

1. Introduction

1.1. Prerequisites



This document assumes you have completed the initial setup of your modem and development kit. If you have not completed those steps, refer to the Skywire® Development Kit User Manual and complete the modem setup before proceeding. If applicable, the modem must already be provisioned. See the Development Kit User Manual for details on how to accomplish these steps.

Note: If your Skywire modem is not listed here, please visit its respective page on www.nimbelink.com for the socket dial procedure.

2. Socket Dial

2.1. Overview

Socket dialing is useful for uploading or downloading information to or from a website or database via HTTP commands. Below is an example for connecting to a server, uploading data, downloading data, and then disconnecting from the server.

2.2. Setup APN (If Applicable)

For the following modems:

- NL-SW-GPRS
- NL-SW-HSPA
- NL-SW-HSPA-B
- NL-SW-LTE-TEUG
- NL-SW-LTE-TNAG
- NL-SW-LTE-TNAG-B
- NL-SW-LTE-TC4EU
- NL-SW-LTE-TC4NAG (AT&T Firmware)
- NL-SW-LTE-TC1NAG (AT&T Firmware)
- NL-SW-LTE-TG1WWG (AT&T Firmware)

type the following command into the terminal program:

```
AT+CGDCONT=1,"IP", "[your apn]"
```

where **[your apn]** is the APN for your carrier, followed by the Enter key.

The terminal should respond with:

```
OK
```

For example, if you are using an AT&T SIM and your APN is:

```
iot0718.com.attz
```

you would type:

```
AT+CGDCONT=1,"IP", "iot0718.com.attz"
```

followed by the Enter key.

For the following modem:

- NL-SW-LTE-TSVG
- NL-SW-LTE-TC4NAG (Verizon Firmware)
- NL-SW-LTE-TC1NAG (Verizon Firmware)
- NL-SW-LTE-TG1WWG (Verizon Firmware)

type the following command into the terminal program:

```
AT+CGDCONT=3,"IP", "[your apn]"
```

where **[your apn]** is the APN for your carrier, followed by the Enter key.

For example, if your Verizon APN is:

```
nimblink.gw12.vzwentp
```

you would type:

```
AT+CGDCONT=3,"IPV4V6", "nimblink.gw12.vzwentp"
```

followed by the Enter key.

2.3. Configure Socket (If Applicable)

For the following modem:

- NL-SW-LTE-TSVG
- NL-SW-LTE-TC4NAG
- NL-SW-LTE-TC1NAG

type the following command into the terminal program:

```
AT#SCFG=1,3,300,90,600,50
```

followed by the Enter key, and the terminal should respond with:

```
OK
```

Since Verizon modems use PDP Context 3, this setup is required to use PDP context 3 with the first socket.

2.4. Activate Skywire Socket

For the following modems:

- NL-SW-1xRTT-A
- NL-SW-1xRTT-S
- NL-SW-1xRTT-V
- NL-SW-EVDO-A
- NL-SW-EVDO-V
- NL-SW-GPRS

- NL-SW-HSPA
- NL-SW-HSPA-B
- NL-SW-LTE-TEUG
- NL-SW-LTE-TNAG
- NL-SW-LTE-TNAG-B
- NL-SW-LTE-TC4EU
- NL-SW-LTE-TC4NAG (AT&T Firmware)
- NL-SW-LTE-TC1NAG (AT&T Firmware)
- NL-SW-LTE-TG1WWG (AT&T Firmware)

type the following command into the terminal program:

AT#SGACT = 1,1

followed by the Enter key, and the terminal should respond with:

#SGACT: "www.xxx.yyy.zzz"

OK

where **www.xxx.yyy.zzz** is the IP address of your Skywire modem.

For the following modem:

- NL-SW-LTE-TSVG
- NL-SW-LTE-TC4NAG (Verizon Firmware)
- NL-SW-LTE-TC1NAG (Verizon Firmware)
- NL-SW-LTE-TG1WWG (Verizon Firmware)

type the following command into the terminal program:

AT#SGACT=3,1

Followed by the Enter key, and the terminal should respond with:

#SGACT: "www.xxx.yyy.zzz"

OK

where **www.xxx.yyy.zzz** is the IP address of your Skywire modem.

2.5. Initiate Socket Dial

In the terminal program, type the following command:

AT#SD=1,0,80,"example.com"

where **1** is the socket you would like to connect with (1 – 6), **0** is the transmission protocol you'd like to use (0 – TCP, 1 – UDP), **80** is the port number you'd like to connect to (1 – 65535), and **example.com** is the host name or IP address you'd like to connect to, followed by the enter key. After a short delay the terminal program should respond with:

CONNECT

and the cursor will be on the next line.

Note: For the socket number, use the socket you set up in Section 2.3 above.

Note: The TCP port (third parameter) that is typically used is port 80, which is the TCP Port for HTTP. Depending on your application, you may use a different port.

2.6. Send Data via HTTP

In order to send data to the website, you can use the HTTP POST command. The syntax of the POST command is as follows:

POST /test/demo_form.asp HTTP/1.1

POST is the HTTP command being issued, **/test/demo_form.asp** is the endpoint on the server, and **HTTP/1.1** is the HTTP version you will be using.

Note: For this step, you will not be able to see the text you are typing. To ensure that the data is accurate, it is recommended to copy and paste in the data. However, this is not necessary.

In the terminal program, enter the POST data you would like to send according to the above syntax. Make sure that the data is formatted to how your server is expecting it. Once you type the data, press CTRL+M CTRL+J CTRL+M CTRL+J. This will enter the two sets of carriage return/new line characters necessary to signal to the server that data transmission is complete. After a short delay, the terminal program should display:

HTTP/1.1 2xx OK

[text response from server]

NO CARRIER

where **2xx** is the successful response code of the HTTP server. **200** is the general HTTP/1.1 response of OK, meaning that the response was received properly. There are derivations of the 2xx response, however. If you receive an unexpected value, please consult the HTTP/1.1 Protocol Documentation at the following URL:

<http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>

2.7. Receive Data via HTTP

In order to receive data from a website, you must use the HTTP GET command. The syntax of the GET command is similar to the POST command, and is as follows:

GET /test/demo_form.asp HTTP/1.1

where **GET** is the HTTP command being issued, **/test/demo_form.asp** is the location of the data, and **HTTP/1.1** is the HTTP version you will be using.

Note: For this step, you will not be able to see the text you are typing. To ensure that the data is accurate, it is recommended to copy and paste in the data. However, the command still works if typed correctly.

In the terminal program, enter the GET data you would like to request according to the above syntax. Make sure that the data is formatted to how your server is expecting it. Once you type the data, press CTRL+M CTRL+J CTRL+M CTRL+J. This will enter the two sets of carriage return/new line characters necessary to signal to the server that data transmission is complete. After a short delay, the terminal program should display:

```
HTTP/1.1 xxx OK
[text response from server]
NO CARRIER
```

where **2xx** is the successful response code of the HTTP server. **200** is the general HTTP/1.1 response of OK, meaning that the response was received properly. There are derivations of the 2xx response, however. If you receive an unexpected value, please consult the HTTP/1.1 Protocol Documentation at the following URL:

<http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html>

2.8. Shutdown the Socket Connection

In the terminal program, type the following command:

```
+++
```

followed by the Enter key, and the terminal should respond with:

```
NO CARRIER
```

This command (three plus signs) is the escape command to suspend the data connection in the socket connection.

To close the socket, type the following command into the terminal program:

```
AT#SH=x
```

where **x** is the socket opened in the previous steps, followed by the enter key, and the terminal should respond with:

```
OK
```

The data connection and socket connection has now been terminated.

3. Working Example 1: dweet.io

3.1. Overview

This section will provide a working example of sending and receiving data via an HTTP transfer using a Skywire NL-SW-LTE-TSVG modem using a 4G Verizon LTE SIM.

dweet.io is a lightweight messaging service specifically designed for IoT (Internet of Things) devices. In addition to being lightweight, dweet.io does not require an account to get up and running. At www.dweet.io, they have an excellent "hello world" example we will be using.

3.2. Get IMEI of Modem

dweet.io requires a unique device name in order to send and receive data. For this, we will be using the IMEI of our modem, which is unique to our modem. To display the IMEI, type the following command into the terminal program:

```
AT+GSN
```

followed by the Enter key, and the terminal will respond with something similar to:

```
XXXXXXXXXXXXXXXXXX
```

```
OK
```

The number should be identical to the IMEI printed on the top of the modem label.

3.3. Initiate Socket Dial

Using the syntax described in Section 2.3, type the following command into the terminal program:

```
AT#SD=1,0,80,"dweet.io"
```

followed by the enter key, and the terminal program should respond with

```
CONNECT
```

In this case, **1** is the socket we are using on the Skywire modem, **0** means we are using the TCP transmission protocol, **80** is the TCP port we are using (TCP port 80 is used for HTTP), and **dweet.io** is the hostname we are connecting to.

3.4. Send Data via HTTP

Using the syntax described in step 2.4, enter the following command into the terminal program:

```
POST /dweet/for/XXXXXXXXXXXXXXXXXX?hello=world HTTP/1.1
```

where the string of **x**'s is your IMEI from setup 3.2, followed by the enter key. Press CTRL+M CTRL+J CTRL+M CTRL+J, and after a short delay the terminal program should respond with something similar to:

```
HTTP/1.1 200 OK
```

```
Access-Control-Allow-Origin: *
```

```
Content-Type: application/json
```

Content-Length: 150
Date: [today's date]
Connection: keep-alive

```
{"this":"succeeded","by":"dweeting","the":"dweet","with":{"thing":"xxxxxxxxxxxxxxxx","created":"[today's date]","content":{"hello":"world"}}}
```

where **xxxxxxxxxxxxxxxx** is your IMEI. By going to the website <https://dweet.io/get/latest/dweet/for/xxxxxxxxxxxxxxxx>, replacing the x's at the end with your IMEI, you will see the following text:

```
{"this":"succeeded","by":"getting","the":"dweets","with":[{"thing":"xxxxxxxxxxxxxxxx","created":"[today's date]","content":{"hello":"world"}}]}
```

This text is the same as the text at the bottom of the terminal response, indicating a successful HTTP transaction.

3.5. Receive Data via HTTP

Using the syntax described in step 2.5, enter the following command into the terminal program:

```
GET /get/latest/dweet/for/xxxxxxxxxxxxxxxx
```

where the string of **x's** is your IMEI from setup 3.2, followed by the enter key. Press CTRL+M CTRL+J CTRL+M CTRL+J, and after a short delay the terminal program should respond with something similar to:

```
HTTP/1.1 200 OK  
Access-Control-Allow-Origin: *  
Content-Type: application/json  
Content-Length: 152  
Date: [today's date]  
Connection: keep-alive
```

```
{"this":"succeeded","by":"getting","the":"dweets","with":[{"thing":"xxxxxxxxxxxxxxxx","created":"[today's date]","content":{"hello":"world"}}]}
```

where **xxxxxxxxxxxxxxxx** is your IMEI.

This indicates that we successfully received the HTTP information.

4. Troubleshooting

4.1 New Line/Carriage Return

If CTRL+M CTRL+J does not insert the necessary new line/carriage return characters, or if your terminal program handles new lines differently, try pressing CTRL+J only instead of CTRL+M CTRL+J

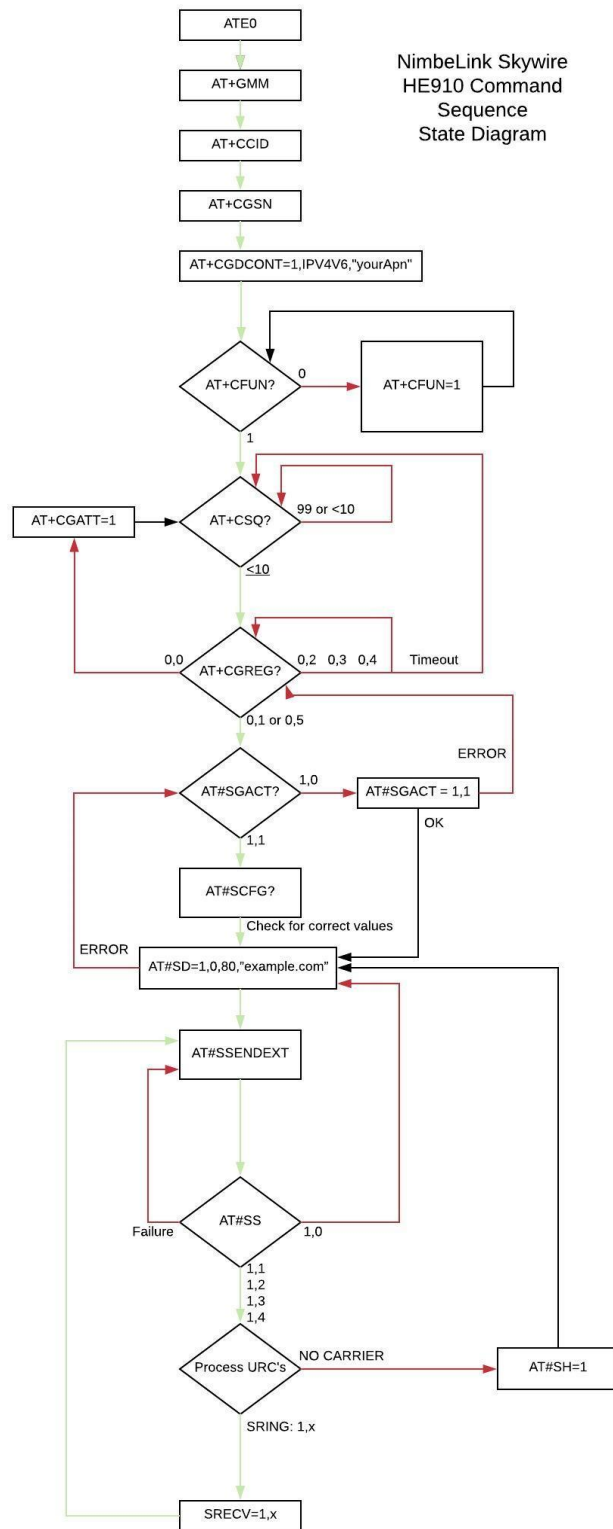
5. Appendix – Socket Dial Flow Charts

5.1. Introduction

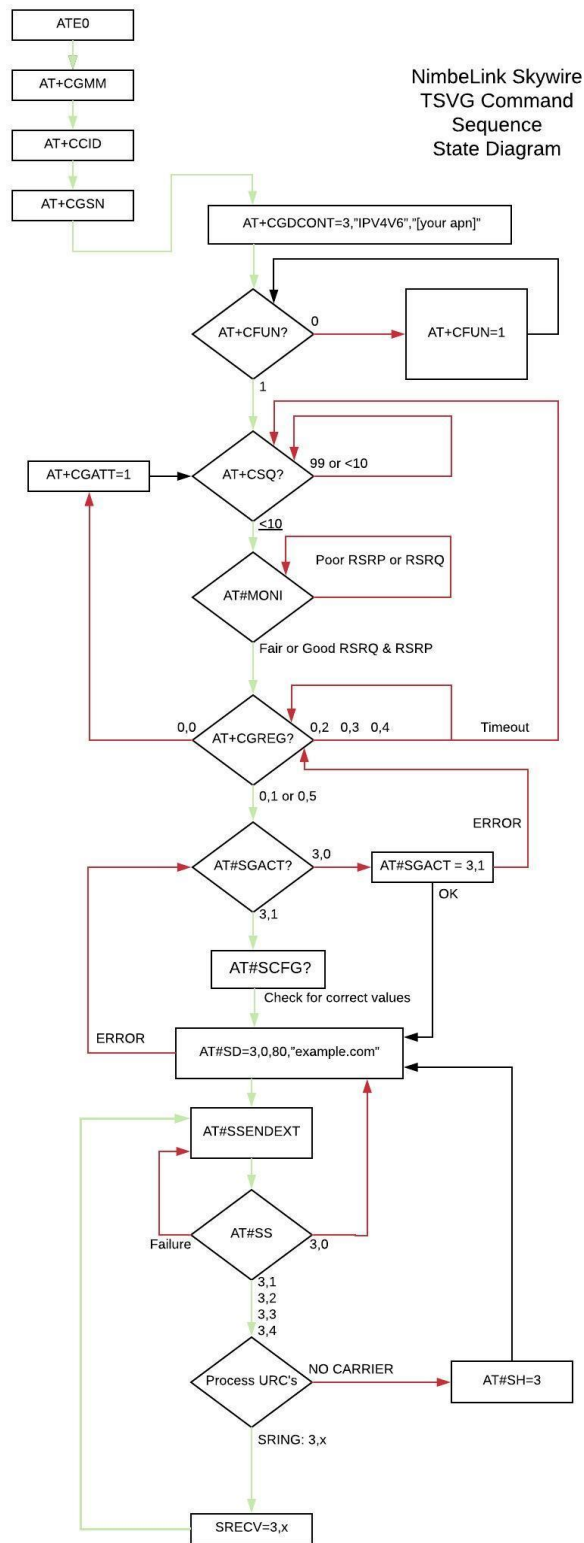
NimbeLink has created example flow charts for the socket dial process, as well as what to do in the event of a failure. NimbeLink recommends at a minimum following these charts and implementing the necessary error handling.

Please see the respective AT command manual for more information on AT commands.

5.2. NL-SW-HSPA Flow Chart



5.3. NL-SW-LTE-TSVG-x Flow Chart



6. Document Version Information

Revision	Description	Date
12	<ul style="list-style-type: none">- Updated Revision Control- Updated to include TG1WWG- Fixed Formatting Errors	10/21/2022