



Cellular Interface Modules

This chapter provides configuration details for the cellular interface modules used in the IR800 series routers.

It is important to understand the architecture of the IR800 series and the relationship between Modems, SIMs, Interface and Controller. The following table helps to illustrate these relationships.

Router	Controller	SIM	Modem Slot	PDN Interface	Line
IR829	0	0 1	0	Cellular 0	3
IR829	0	0 1	0	Cellular 1	8
IR829 (dual modem) *	0	0	0	Cellular 0/0	3
IR829 (dual modem) *	0	0	0	Cellular 0/1	8
IR829 (dual modem) *	1	1	1	Cellular 1/0	9
IR829 (dual modem) *	1	1	1	Cellular 1/1	15
IR809	0	0 1	0	Cellular 0	3
IR809	0	0 1	0	Cellular 1	8



Note Check the [Product Marketing Data Sheet](#) for updated modem information.

With the introduction of the next generation SKUs, some functionality has changed. Refer to the following table for details.

Description	IR829GW-[LA/GA/NA/VZ]-*K9	IR829-2LTE-EA-*K9
North American	Yes	Yes
APJC	Yes	No

Description	IR829GW-[LA/GA/NA/VZ]-*K9	IR829-2LTE-EA-*K9
EMER	Yes	Yes
EMEA	Yes	Yes
2G Support	Yes	No
3G Support	Yes	Yes
LTE Support	Yes	Yes
GPS	Yes	Yes Note Only available from the first LTE Modem.
Wi-Fi (2.4/5 GHZ)	2.4 GHz and 5GHz use separate antenna connector	2.4 GHz + 5GHz coexist on the same antenna connector
Dual SIM	Yes	No
Band 30	No	No
LTE category supported	cat4	cat4

This chapter contains the following sections:

- [Cellular Interface, on page 2](#)
- [Radio Frequency Band Select, on page 15](#)
- [Low Power Mode, on page 16](#)
- [Enhancement to Modem Crash Action, on page 16](#)
- [IR800 Cellular Technology Selection, on page 17](#)
- [GPS, on page 20](#)
- [Troubleshooting the Cellular Interface, on page 23](#)

Cellular Interface

The Cisco IR800 series Industrial routers use the Sierra Wireless MC73XX and MC74XX series modems supporting MIMO on LTE. WCDMA UMTS HSPA DC-HSPA+ is diversity only, without MIMO.

Installation of the SIM card(s) and antennas is covered in the respective Hardware Installation Guides under the Cisco 800 Series Industrial Integrated Services Routers page:

<http://www.cisco.com/c/en/us/support/routers/800-series-industrial-routers/tsd-products-support-series-home.html>

The software download page can be found here:

<https://software.cisco.com/download/navigator.html?mdfid=286288566&flowid=76082>

The Firmware Upgrade Guide for Cellular Modems can be found here:

http://www.cisco.com/c/en/us/td/docs/routers/access/interfaces/firmware/Firmware_Upgrade.html

Cisco 4G LTE Software Installation Guide

<http://www.cisco.com/c/en/us/td/docs/routers/access/interfaces/software/feature/guide/EHWIC-4G-LTESW.html>

After installing the SIM card(s) and antennas, check the cellular hardware, radio, network and SIM (Unlock SIM card if necessary).

4G LTE Dual SIM

Dual Subscriber Identity Module (SIM) provides reliability and multihoming capabilities over LTE and HSPA-based networks. With two LTE modems, the IR829 enables concurrent connectivity to two cellular networks for high reliability, enhanced data throughputs, load balancing and differentiated services.



Note Dual SIM active/backup mode is supported only on single LTE models of the IR829.

The following features are provided:

- The two SIMs operate in active/backup mode on the single LTE models of the IR829, and active/active mode with each of the two SIMs assigned to a specific cellular radio on the dual LTE models. Both mobile provider networks must be supported by the given IR829 SKU, and it must be in an applicable region.
- By default, SIM slot 0 is the primary, and SIM slot1 is the backup. Behavior may be changed using the **lte sim primary** command.
- Profiles for each SIM are assigned by using the **lte sim profile** command. Each SIM has an associated Internet profile and an IMS profile in the CLI.
- Dual-SIM behavior is managed under Cellular 0 CLI configuration.
- The fail over occurs when there is no signal from the current carrier, and generally happens depending on the fail over timer value that is set. The default value is 5 minutes. The range is from 0-7 minutes..
- Dual active LTE radios providing Multi-carrier support for active and backup use cases. Newer cellular modems have been added (MC74xx) with FDD/TDD LTE on LA and EA 829 models.



Note The 7455 modems do not support dual SIM capabilities.

AutoSim and Firmware Based Switching

The advantages of the AutoSim feature are:

- Ease of Ordering Carrier Specific SKUs
- Quicker failover times in dual-sim deployments
- Ease of switchover from other service providers to Telstra network

Auto-SIM is supported in Sierra wireless firmware Version 02.20.03. A new CLI is added in the cellular controller to enable/disable Auto-SIM. The modem in Auto-SIM mode selects the right carrier firmware after a SIM slot switch and an automatic modem reset. Auto-SIM is supported on the MC7455, MC7430, EM7430,

and EM7455 modems. During bootup, if the Auto-SIM configuration on the modem doesn't match to the IOS configuration, the corresponding Auto-SIM or manual mode is pushed to the modem.

After an Auto-SIM configuration change, the modem is automatically reset; the default is "auto-sim" enabled.

Enable Auto-SIM:

```
router(config)#controller cellular <slot>
router(config-controller)#lte firmware auto-sim #default is auto-sim enabled
```



Note After enabling auto-sim, wait for 5 minutes until the radio comes up. Once the radio is up, issue a modem power-cycle and wait for 3 minutes for the radio to come up again. Modem Power-Cycle is mandatory for auto-sim configuration to take effect.

Disable Auto-SIM:

```
router(config)#controller cellular <slot>
router(config-controller)#no lte firmware auto-sim
```



Note After disabling auto-sim, wait for 5 minutes until the radio comes up. Once the radio is up, issue a modem power-cycle and wait for 3 minutes for the radio to come up again. Modem Power-Cycle is mandatory for auto-sim configuration to take effect.

If Auto-SIM is disabled and the modem is in manual mode, select a carrier with a new exec CLI:

```
cellular lte firmware-activate <firmware-index>
```

The following CLI example shows the firmware-index of the carrier in the modem:

```
router#show cellular <slot> firmware
```

For additional information, see the following guide: https://www.cisco.com/c/en/us/td/docs/routers/access/interfaces/NIM/software/configuration/guide/4GLTENIM_SW.html

Dual Radio Configuration and Single Radio Configuration

The following examples are of an IR800 cellular configuration using dual modems. A single modem example will look much the same, without the *Cellular1/0* and *Cellular1/1* entries.

```
DUAL-Modem> enable
DUAL-Modem# show ip int brief
```

Interface	IP-Address	OK?	Method	Status	Protocol
GigabitEthernet0	unassigned	YES	NVRAM	administratively down	down
GigabitEthernet1	unassigned	YES	unset	down	down
GigabitEthernet2	unassigned	YES	unset	down	down
GigabitEthernet3	unassigned	YES	unset	down	down
GigabitEthernet4	unassigned	YES	unset	down	down
Wlan-GigabitEthernet0	unassigned	YES	unset	up	up
Async0	unassigned	YES	unset	up	down
Async1	unassigned	YES	unset	up	down
GigabitEthernet5	unassigned	YES	NVRAM	administratively down	down
Cellular0/0	192.168.43.237	YES	IPCP	up	up
Cellular1/0	10.61.25.231	YES	IPCP	up	up
Second Modem					
Cellular0/1	unassigned	YES	TFTP	down	down

```

Cellular1/1                unassigned      YES TFTP   down          down
Second Modem
Vlan1                      unassigned      YES unset   up            up
wlan-ap0                   unassigned      YES NVRAM   up            up
DUAL-Modem# show running-config

```

```

Building configuration...
Current configuration : 4021 bytes
!
! Last configuration change at 18:31:06 UTC Mon Oct 24 2016
!
version 15.6
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
service internal
!
hostname DUAL-Modem
!
boot-start-marker
boot system flash:/ir800-universalk9-mz.SPA.156-3.M0a
boot-end-marker
!
no aaa new-model
ethernet lmi ce
service-module wlan-ap 0 bootimage autonomous
!
ignition off-timer 900
!
ignition undervoltage threshold 9
!
no ignition enable
!
no ip domain lookup
ip inspect WAAS flush-timeout 10
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
chat-script lte "" "AT!CALL" TIMEOUT 20 "OK"
!
license udi pid IR829-2LTE-EA-BK9 sn FGL2032219N
!
redundancy
notification-timer 120000
controller Cellular 0
lte sim data-profile 3 attach-profile 1

    #When using Verizon, use data profile 3 and attach to profile 1
    #When using AT&T, use data profile 1 and attach to profile 1

lte modem link-recovery rssi onset-threshold -110
lte modem link-recovery monitor-timer 20
lte modem link-recovery wait-timer 10
lte modem link-recovery debounce-count 6
!
controller Cellular 1
lte modem link-recovery rssi onset-threshold -110
lte modem link-recovery monitor-timer 20
lte modem link-recovery wait-timer 10
lte modem link-recovery debounce-count 6
interface GigabitEthernet0
no ip address

```

```

shutdown
!
interface GigabitEthernet1
no ip address
!
interface GigabitEthernet2
no ip address
!
interface GigabitEthernet3
no ip address
!
interface GigabitEthernet4
no ip address
!
interface Wlan-GigabitEthernet0
no ip address
!
interface GigabitEthernet5
no ip address
shutdown
duplex auto
speed auto
!
interface Cellular0/0
    #Both interfaces need to be configured in the IOS software

ip address negotiated
ip virtual-reassembly in
encapsulation slip
load-interval 30
dialer in-band
dialer string lte
dialer-group 1
no peer default ip address
async mode interactive
routing dynamic
!
interface Cellular1/0
    #Both interfaces need to be configured in the IOS software
ip address negotiated
ip virtual-reassembly in
encapsulation slip
load-interval 30
dialer in-band
dialer string lte
dialer-group 1
no peer default ip address
async mode interactive
routing dynamic
!
interface Cellular0/1
no ip address
encapsulation slip
!
interface Cellular1/1
no ip address
encapsulation slip
!
interface wlan-ap0
no ip address
!
interface Vlan1
no ip address
!

```

```

interface Async0
no ip address
encapsulation scada
!
interface Async1
no ip address
encapsulation scada
!
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 Cellular1/0
ip route 8.8.8.8 255.255.255.255 Cellular0/0

Route values added
!
dialer-list 1 protocol ip permit
ipv6 ioam timestamp
!
access-list 1 permit any
!
control-plane
!
!
line con 0
stopbits 1
line 1 2
stopbits 1
line 3
script dialer lte
no exec
transport preferred lat pad telnet rlogin lapb-ta mop udptn v120 ssh
transport output lat pad telnet rlogin lapb-ta mop udptn v120 ssh
rxspeed 150000000
txspeed 50000000
line 4
no activation-character
no exec
transport preferred none
transport input all
transport output lat pad telnet rlogin lapb-ta mop udptn v120 ssh
!
line 8
script dialer lte
no exec
transport preferred lat pad telnet rlogin lapb-ta mop udptn v120 ssh
transport output lat pad telnet rlogin lapb-ta mop udptn v120 ssh
rxspeed 150000000
txspeed 50000000
line 9
script dialer lte
no exec
transport preferred lat pad telnet rlogin lapb-ta mop udptn v120 ssh
transport input all
transport output lat pad telnet rlogin lapb-ta mop udptn v120 ssh
rxspeed 236800
txspeed 118000
line 15
no exec
transport preferred lat pad telnet rlogin lapb-ta mop udptn v120 ssh
transport output lat pad telnet rlogin lapb-ta mop udptn v120 ssh

```

```

rxspeed 236800
txspeed 118000
line 1/3 1/6
transport preferred none
transport output none
stopbits 1
line vty 0 4
login
transport input none
!
no scheduler max-task-time
!!
End

```

Test the modem configuration with a ping command:

```

DUAL-Modem# ping 8.8.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 30/88/292 ms
DUAL-Modem#

```

The following two examples show a Verizon profile followed by an AT&T profile.

Verizon Profile

```

DUAL-Modem# show cellular 0/0 profile

Profile 1 = INACTIVE **
-----
PDP Type = IPv4v6
Access Point Name (APN) = vzwims
Authentication = None
Profile 2 = INACTIVE
-----
PDP Type = IPv4v6
Access Point Name (APN) = vzwadmin
Authentication = None
Profile 3 = ACTIVE*

Profile 3 is used for Verizon
-----
PDP Type = IPv4v6
PDP address = 166.140.43.237
Access Point Name (APN) = we01.VZWSTATIC
Authentication = None
    Primary DNS address = 198.224.173.135
    Secondary DNS address = 198.224.174.135
Profile 4 = INACTIVE
-----
PDP Type = IPv4v6
Access Point Name (APN) = vzwapp
Authentication = None
Profile 5 = INACTIVE
-----
PDP Type = IPv4v6
Access Point Name (APN) = vzw800
Authentication = None
Profile 6 = INACTIVE

```

```

-----
PDP Type = IPv4v6
Access Point Name (APN) = vzwenterprise
Authentication = None
    * - Default profile
    ** - LTE attach profile

```

AT&T Profile

```

DUAL-Modem# show cellular 1/0 profile

Profile 1 = ACTIVE* **

Profile 1 is used for AT&T
-----
PDP Type = IPv4
PDP address = 10.61.25.231
Access Point Name (APN) = m2m.com.attz
Authentication = None
    Primary DNS address = 8.8.8.8
    Secondary DNS address = 8.8.4.4
    * - Default profile
    ** - LTE attach profile
DUAL-Modem# show cellular 0/0 hardware
Modem Firmware Version = SWI9X30C_02.20.03.00
Modem Firmware built = 2016/06/30 10:54:05
Hardware Version = 1.0
Device Model ID: MC7455MOBILE
International Mobile Subscriber Identity (IMSI) = 311480166946902
International Mobile Equipment Identity (IMEI) = 352009080050110
Integrated Circuit Card ID (ICCID) = 89148000001653263375
Mobile Subscriber Integrated Services
Digital Network-Number (MSISDN) = 6692200807
Modem Status = Online
Current Modem Temperature = 34 deg C
PRI SKU ID = 1103084, PRI version = 002.024, Carrier = Verizon
Carrier identified as Verizon
OEM PRI version = 000.001

```

Creating a Cellular Profile for Verizon.

```

DUAL-Modem# cellular 0/0 lte profile create 3 we01.VZWSTATIC
Warning: You are attempting to modify a currently ACTIVE data profile.

This is not recommended and may affect the connection state
PDP Type = IPv4v6
Access Point Name (APN) = we01.VZWSTATIC
Authentication = NONE
Profile 3 already exists with above parameters. Do you want to overwrite? [confirm] <return>
Profile 3 will be overwritten with the following values:
PDP type = IPv4
APN = we01.VZWSTATIC
Authentication = NONE
Are you sure? [confirm] <return>
Profile 3 written to modem
DUAL-Modem#
Enter configuration commands, one per line. End with CNTL/Z.
DUAL-Modem(config)# controller cellular 0
DUAL-Modem(config-controller)# lte sim data-profile 3 attach-profile 1
DUAL-Modem(config-controller)#
DUAL-Modem# conf t

```

```

Enter configuration commands, one per line.  End with CNTL/Z.
DUAL-Modem(config)# controller cellular 0
DUAL-Modem(config-controller)# lte sim data-profile 3 attach-profile 1
DUAL-Modem(config-controller)# end
DUAL-Modem#
DUAL-Modem# show

*Oct 24 19:43:44.841: %SYS-5-CONFIG_I: Configured from console by consolecell
DUAL-Modem# show cellular 1/0 profile
Profile 1 = ACTIVE* **
-----
PDP Type = IPv4
PDP address = 10.61.185.213
Access Point Name (APN) = m2m.com.attz
Authentication = None
    Primary DNS address = 8.8.8.8
    Secondary DNS address = 8.8.4.4
  * - Default profile
  ** - LTE attach profile

```

Creating a Cellular Profile for AT&T

```

DUAL-Modem# cellular 1/0 lte profil create 1 m2m.com.attz
Warning: You are attempting to modify a currently ACTIVE data profile.

This is not recommended and may affect the connection state
PDP Type = IPv4
Access Point Name (APN) = m2m.com.attz
Authentication = NONE
Profile 1 already exists with above parameters. Do you want to overwrite? [confirm] <return>
Profile 1 will be overwritten with the following values:
PDP type = IPv4
APN = m2m.com.attz
Authentication = NONE
Are you sure? [confirm] <return>
Profile 1 written to modem
DUAL-Modem#
DUAL-Modem# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
DUAL-Modem(config)# controller cellular 1
DUAL-Modem(config-controller)#
DUAL-Modem(config-controller)# lte sim data-profile 1 attach-profile 1

Note: Please issue a modem reset for the modified attach-profile to take effect.
DUAL-Modem(config-controller)# end
DUAL-Modem#

```

Controller Cellular 0 and NAT Configuration

Controller Cellular 0 is configured with default parameters. If a profile different from Profile 1 is set-up, it must be attached to controller cellular 0.

If the SIM in slot #1 must be used as primary, it is done under controller cellular 0

Procedure

Step 1 Show the controller cellular 0

Example:

```
IR800#show run | begin controller
controller Cellular 0
  lte sim data-profile 1 attach-profile 1 slot 0 !
Value set-up for configuration example
  lte sim max-retry 0
  lte failovertime 0
  lte modem link-recovery rssi onset-threshold -110
  lte modem link-recovery monitor-timer 20
  lte modem link-recovery wait-timer 10
  lte modem link-recovery debounce-count 6
!
```

Step 2 If the cellular interface obtains an IPv4 private address, NAT should be configured.

Example:

```
IR800#conf term
Enter configuration commands, one per line. End with CNTL/Z.
IR800(config)#inter cellular 0
IR800(config-if)#ip nat outside
IR800(config)#inter vlan 4
IR800(config-if)#ip nat inside
IR800(config)#access-list 10 permit 10.20.20.0 0.0.0.255
!
IPv4 subnet to be NATed
IR800(config)# ip nat inside source list 10 interface Cellular0 overload
!
NAT interface association
```

Step 3 Once the Cellular configuration is done, ping a well-known IP address to test the connectivity.

Example:

```
IR800#ping 8.8.8.8

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 340/472/740 ms
IR800#
```

Step 4 Attached Cellular 0 profile must become “active” and “connection” shows IP address and traffic.

Example:

```
IR800#show cellular 0 profile
Profile 1 = ACTIVE* **
-----
PDP Type = IPv4
PDP address = 10.60.159.255
Access Point Name (APN) = LTE
Authentication = None
  Primary DNS address = 212.27.40.240
  Secondary DNS address = 212.27.40.241
  * - Default profile
  ** - LTE attach profile
Configured default profile for active SIM 0 is profile 1.
IR800#show cellular 0 connection
Profile 1, Packet Session Status = ACTIVE
Cellular0:
```

```
Data Transmitted = 700 bytes, Received = 600 bytes
IP address = 10.60.159.255
Primary DNS address = 212.27.40.240
Secondary DNS address = 212.27.40.241
Profile 2, Packet Session Status = INACTIVE
```

What to do next

Use the `show interface cellular 0` command to display the negotiated IP address if operational.

```
IR800#show interfaces cellular 0
Cellular0 is up, line protocol is up
  Hardware is 4G WWAN Modem - Global (Europe & Australia) Multimode LTE/DC-HSPA+/HSPA+/HSPA/U

  Internet address is 10.123.161.59/32
  MTU 1500 bytes, BW 384 Kbit/sec, DLY 100000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation SLIP, loopback not set
  Keepalive not supported
  Last input 00:22:41, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/10 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    12 packets input, 1128 bytes, 0 no buffer
    Received 0 broadcasts (0 IP multicasts)
    0 runs, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    51 packets output, 3364 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
    0 unknown protocol drops
    0 output buffer failures, 0 output buffers swapped out
    0 carrier transitions
  DCD=up DSR=up DTR=up RTS=up CTS=up
IR800#
```

If the negotiated IP address is not operational:

```
IR800#show interfaces cellular 0
Cellular0 is up (spoofing), line protocol is up (spoofing)
  Hardware is 4G WWAN Modem - Global (Europe & Australia) Multimode LTE/DC-HSPA+/HSPA+/HSPA/U

  Internet address will be assigned dynamically by the network
```

Other Useful Commands

```
IR800# show cell 0 hardware
Modem Firmware Version = SWI9X15C_05.05.58.00
Modem Firmware built = 2015/03/04 21:30:23
Hardware Version = 1.0
Device Model ID: MC7304
Package Identifier ID: 1102029_9903299_MC7304_05.05.58.00_00_Cisco_005.010_000
International Mobile Subscriber Identity (IMSI) = 208150103324395
International Mobile Equipment Identity (IMEI) = 352761060206340
```

```

Integrated Circuit Card ID (ICCID) = 8933150112100222053
Mobile Subscriber Integrated Services
Digital Network-Number (MSISDN) = 33695764790
Current Modem Temperature = 47 deg C
PRI SKU ID = 9903299, PRI version = 05.10, Carrier = 1

IR800# show cell 0 security
Active SIM = 0 !
SIM slot #0 active
SIM switchover attempts = 0
Card Holder Verification (CHV1) = Disabled
SIM Status = OK
SIM User Operation Required = None
Number of CHV1 Retries remaining = 3

IR800# cellular 0 lte sim unlock XXXX
!
XXXX = PIN code

IR800# show cell 0 radio
Radio power mode = ON
Channel Number = 3037
Current Band = Unknown
Current RSSI(RSCP) = -99 dBm
Current ECIO = -10 dBm
Radio Access Technology(RAT) Preference = AUTO
Radio Access Technology(RAT) Selected = UMTS ( UMTS/WCDMA )

IR800# show cell 0 network
Current System Time = Sat Oct 10 9:12:59 2015
Current Service Status = Normal
Current Service = Packet switched
Current Roaming Status = Home
Network Selection Mode = Automatic
Network = LTE
Mobile Country Code (MCC) = 208
Mobile Network Code (MNC) = 15
Packet switch domain(PS) state = Attached
Location Area Code (LAC) = 3910
Cell ID = 222094374

IR800# show cell 0 all

```



Note The output to the **show cell 0 all** command is extensive, and omitted from this guide for brevity.

Accessing 4G Modem AT Commands



Note A password must be added to the line configuration for security.

Get the line number associated to Cellular 0:

```

IR800#show line
  Tty Line Typ  Tx/Rx   A Modem  Roty AccO AccI  Uses  Noise Overruns  In
  I    3     3 TTY   -      -       -    -    -    1      0    4/0    Ce0

```

Use one of the IR800 IP address along with 2000 + line number (2003)

```
IR800#10.15.15.1 2003
Trying 10.15.15.1, 2003 ... Open
```

Execute the 4G modem AT commands, for example AT!GSTATUS?:

```
AT!GSTATUS?
!GSTATUS:
Current Time: 213353 Temperature: 38
Bootup Time: 0 Mode: ONLINE
System mode: WCDMA PS state: Attached
WCDMA band: WCDMA 900
WCDMA channel: 3037
GMM (PS) state:REGISTERED NORMAL SERVICE
MM (CS) state: IDLE NORMAL SERVICE
WCDMA L1 state:L1M_PCH_SLEEP LAC: 0F46 (3910)
RRC state: DISCONNECTED Cell ID: 0D3CE428 (222094376)
RxM RSSI C0: -90 RxD RSSI C0: -106
RxM RSSI C1: -106 RxD RSSI C1: -106
```

Disconnect using “SHIFT+CONTROL+6+x”, then confirm:

```
IR800#disc
Closing connection to 10.2.2.2 [confirm]enter
IR800#
```

Checking 4G Modem Firmware through AT Commands

To check the IR800 4G modem firmware, execute the 4G modem AT commands after connecting to the modem. The following example is for an IR809G-LTE-GA-K9 loaded with FW-MC7304-LTE-GB Global firmware.



Note On the IR809, the PRI SKU ID= 9903299 is not representative of the GB firmware

```
at!priid?
PRI Part Number: 9903299
Revision: 05.10
Carrier PRI: 9999999_9902674_SWI9X15C_05.05.58.00_00_GENEU-4G_005.026_000
OK
at!package?
1102029_9903299_MC7304_05.05.58.00_00_Cisco_005.010_000
at!gobiimpref?
!GOBIIMPREF:
preferred fw version: 05.05.58.00
preferred carrier name: GENEU-4G
preferred config name: GENEU-4G_005.026_000
current fw version: 05.05.58.00
current carrier name: GENEU-4G
current config name: GENEU-4G_005.026_000
```

Radio Frequency Band Select

This new feature allows the user to configure and lock down the modem to a specific RF band, or set of bands. The preference can be set to be equal to, or a sub-set of the capability supported by the modem/carrier combination.

The following examples show the controller configuration commands:

```
router# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
router(config)#controller cell interface number

router(config-controller)#lte modem ?
    band-select      Modem band select
    dm-log            Modem DM logging configuration
    fota-poll-timer   Set poll timer for AVMS to do Firmware upgrade over the air
    link-recovery     Cellular Link Recovery
    mtu               Modem mtu
    nas-log           Modem NAS logging configuration

router(config-controller)#lte modem band-select ?
    all-lte-only      Choose all LTE bands only
    all-nonlte-only   Choose all non-LTE bands only
    band-indices      Specify the lte and non-lte band indices

router(config-controller)#lte modem band-select band-indices ?
    WORD              Band index(es) in string format "<band index#>, <band index#>, ...".
                     (supported band indices are listed under 'show cellular radio band'.)

router(config-controller)#lte modem band-select band- indices "2 4 5" ?
    slot              primary SIM slot

router(config-controller)#lte modem band-select band- indices "2 4 5" slot ?

    <0-1>              Slot number

router(config-controller)#lte modem band-select band- indices "2 4 5" slot 0

router#show run | sec controller
controller Cellular 0
    lte sim max-retry 0
    lte failovertime 4
    lte modem dm-log rotation
    lte modem link-recovery disable
    lte modem band-select band- indices "2,4,5" slot 0
```

The following examples show the controller show commands:

```
router#show cellular interface number radio ?

    band              Show Radio band settings
    history            Show Radio history in graph format
    |                 Output modifiers
    <cr>               <cr>

router#show cellular interface number radio band
```

```

LTE bands supported by modem:
- Bands 2 4 5 12.
LTE band Preference settings for the active sim(slot 0):
- Bands 2 4 5 12.
Non-LTE bands supported by modem:
Index:
  88 - WCDMA US PCS 1900 band
  90 - WCDMA US 1700 band
  91 - WCDMA US 850 band
Non-LTE band Preference settings for the active sim(slot 0):
Index:
  88 - WCDMA US PCS 1900 band
  90 - WCDMA US 1700 band
  91 - WCDMA US 850 band

IR807#show run | sec controller
controller Cellular 0
no lte gps enable
lte modem crash-action boot-and-hold
lte modem fota-poll-timer 15
lte modem mtu 1700
lte modem link-recovery disable
IR800#

```

Low Power Mode

This feature provides the reason for the modem going into a low power mode if the situation ever occurs. It uses the device power control information provided by the modem. A new CLI has been implemented **show cellular <interface> radio details**.

```

router# show cellular <interface number> radio

Radio power mode = OFF, Reason = User Request
Channel Number = 0
Current Band = Unknown
Current RSSI = -128 dBm
Current ECIO = -2 dBm
Radio Access Technology(RAT) Preference = AUTO
Radio Access Technology(RAT) Selected = AUTO

router# show cellular <interface number> radio details

Radio turned off under cellular controller configuration.
router#

```

Note: In the above **show cellular <interface number> radio** output, the Radio power mode shows **OFF** because the user has turned the radio off by choice. In all other cases, when the radio goes to Low Power mode, you will see the display Radio power mode = **low power**.

Enhancement to Modem Crash Action

If the modem corresponding to the cellular interface crashes, the modem will reset itself and come back up. However, in order to debug the cause of the crash, a full crash dump can be captured on the modem. The steps to capture the crashdump are outlined in:

:
<https://www.cisco.com/c/en/us/support/docs/interfaces-modules/lte-wireless-wan-interfaces/200463-Generate-4G-modem-crash-dump.html>
 or

<https://www.cisco.com/c/en/us/td/docs/routers/access/800/819/user/guide/3G4G-enhancements-userguide.html#pgfid-1076594>

A new CLI has been added to set the crash action on the modem upon a crash. The CLI is **lte modem crash-action ?**. The device can be set to either reset, or to boot and hold.

The following example shows the new functionality of the configuration CLI:

```
Router(config-controller)#lte modem crash-action ?
boot-and-hold  → Remain in crash state
reset          → reset the modem on crash
```

This CLI will set the flag to either 1 or 0 for reset and boot and hold respectively. This is the same as AT command **at!eroption= 0 / 1**

The following example shows the new functionality of the exec CLI:

```
router#show cellular <your interface> logs modem-crash-action
Current modem crash action: Reset
```

This CLI will show the current state the modem is set to. This is the same as AT command **at!eroption=?**.

IR800 Cellular Technology Selection

The cellular interface supports a seamless hand off between LTE and 3G networks when the LTE cell becomes weak in certain spots and vice versa. But it may also be disable to lock the cellular interface in a given technology, for example. LTE.

The cellular interface supports 3G and 2.5G technologies. The IOS CLI can be used to select a particular technology that is most desirable in your local zone.

Use the cellular 0 lte technology command:

```
IR829# cellular 0 lte technology ?
!
Blue
values available on Global SKU

auto    Automatic LTE Technology Selection
cdma-lxrtt  CDMA 1xRTT
cdma-evdo   CDMA EVDO Rev A
cdma-hybrid HYBRID CDMA

gsm       GSM
lte       LTE
umts      UMTS
```



Note The default technology type selection is **auto**, and it is recommended to be used at all times. Although **gsm** and **umts** are part of the selection, the modem firmware does not support them on gsm/umts network. They will be used as **lte** selection on a Verizon network.

Show the completed configuration: (output edited for brevity)

```

IR800#show run
Building configuration...
Current configuration : 4365 bytes
!
! Last configuration change at 09:53:09 UTC Sat Oct 10 2015 by cisco
!
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname IR800
!
boot-start-marker
boot system flash:/ir800-universalk9-mz.SPA.155-3.M0a
boot-end-marker
!
enable password cisco
!
aaa new-model
!
aaa session-id common
ethernet lmi ce
!
ip dhcp pool GuestOS
network 10.16.16.0 255.255.255.0
default-router 10.16.16.1
dns-server 8.8.8.8
!
ip domain name local.cisco.com
ip cef
ipv6 unicast-routing
ipv6 cef
!
multilink bundle-name authenticated
!
chat-script LTE "" "AT!CALL" TIMEOUT 20 "OK"
!
license udi pid IR809G-LTE-GA-K9 sn JMX1915X00Q
license accept end user agreement
license boot module ir800 technology-package securityk9
license boot module ir800 technology-package datak9
!
username cisco password 0 cisco
!
redundancy
!
controller Cellular 0
lte sim data-profile 1 attach-profile 1 slot 0
lte sim max-retry 0
lte failovertimer 0
lte modem link-recovery rssi onset-threshold -110
lte modem link-recovery monitor-timer 20
lte modem link-recovery wait-timer 10
lte modem link-recovery debounce-count 6
!
interface GigabitEthernet0
description backhaul
ip address dhcp
duplex auto
speed auto

```

```
    ipv6 address autoconfig default
!
interface GigabitEthernet1
    no ip address
    shutdown
    duplex auto
    speed auto
!
interface GigabitEthernet2
    ip address 10.16.16.1 255.255.255.0
    duplex auto
    speed auto
    ipv6 address autoconfig
!
interface Cellular0
    ip address negotiated
    encapsulation slip
    dialer in-band
    dialer idle-timeout 0
    dialer string LTE
    dialer-group 1
    async mode interactive
!
interface Cellular1
    no ip address
    encapsulation slip
!
interface Async0
    no ip address
    encapsulation scada
!
interface Async1
    no ip address
    encapsulation scada
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 Cellular0
ip ssh time-out 60
!
dialer-list 1 protocol ip permit
!
control-plane
!
line con 0
    stopbits 1
line 1 2
    stopbits 1
line 3
    script dialer LTE
    modem InOut
    no exec
    transport preferred lat pad telnet rlogin lapb-ta mop udptn v120 ssh
    transport input telnet
    transport output lat pad telnet rlogin lapb-ta mop udptn v120 ssh
    speed 384000
line 8
    script dialer LTE
    modem InOut
    no exec
    transport preferred lat pad telnet rlogin lapb-ta mop udptn v120 ssh
```

```

transport output lat pad telnet rlogin lapb-ta mop udptn v120 ssh
speed 384000
line 1/3 1/6
transport preferred none
transport output none
stopbits 1
line vty 0 4
password cisco
transport input telnet ssh
!
no scheduler max-task-time
!
end
IR800#

```

GPS

The IR800 series can be configured to enable real-time location tracking of remote assets and geo-fence when used with IOT Field Network Director. Field Network Director receives GPS data directly from IOS, not NMEA.

Key Points:

- GPS must be configured under *controller cellular 0*.
- GPS can be assigned to Cellular AUX antenna.
- GPS data can be seen locally, or data stream can be forwarded to applications, i.e. RUBAN.



Note On the IR829 dual-LTE model, GPS can only be configured on cellular 0/0.

For information about the GPS LED indications and locations of the GPS connectors, see [IR829 Product Overview](#) and [IR809 Product Overview](#).

To configure GPS on the IR800 series, refer to the following examples.

```

IR829# conf term
IR829(config)#controller cellular 0
IR829(config-controller)#lte gps ?
    enable    enable GPS feature
    mode      select GPS mode
    nmea      enable NMEA data
IR829(config-controller)#lte gps mode standalone

```

```

IR829(config-controller)#lte gps nmea ip
IR829#show cellular 0 gps

```

```

GPS Info
-----
GPS Feature: enabled
GPS Port Selected: Dedicated GPS port
GPS State: GPS enabled
GPS Mode Configured: standalone
Latitude: 48 Deg 38 Min 31.2114 Sec North
Longitude: 2 Deg 13 Min 47.3992 Sec East
Timestamp (GMT): Wed Jul 22 08:05:28 2015

```

```

Fix type index: 0, Height: 94 m
Satellite Info
-----
Satellite #14, elevation 28, azimuth 310, SNR 31 *
Satellite #15, elevation 22, azimuth 171, SNR 39 *
Satellite #17, elevation 25, azimuth 45, SNR 34 *
Satellite #18, elevation 8, azimuth 248, SNR 25
Satellite #22, elevation 12, azimuth 281, SNR 24
Satellite #24, elevation 78, azimuth 90, SNR 35 *
Satellite #25, elevation 23, azimuth 241, SNR 27
Satellite #1, elevation 0, azimuth 0, SNR 0
Satellite #2, elevation 0, azimuth 0, SNR 0
Satellite #6, elevation 6, azimuth 85, SNR 0
Satellite #12, elevation 62, azimuth 241, SNR 0
Satellite #26, elevation 0, azimuth 0, SNR 0
Satellite #29, elevation 0, azimuth 0, SNR 0
IR829#

```

You can also configure IOS so that GPS can be streamed to another destination (port or address).

For example:

```

IR829#conf t
Enter configuration commands, one per line. End with CNTL/Z.
IR829(config)#controller cellular 0
IR829(config-controller)#lte gps nmea ?
    ip      NMEA over IP interface
    serial  NMEA over serial interface
IR829(config-controller)#lte gps nmea ip ?
    udp     UDP Transport
    <cr>
IR829(config-controller)#lte gps nmea ip udp ?
    A.B.C.D Source address
IR829(config-controller)#lte gps nmea ip udp 10.3.4.5 ?
    A.B.C.D Destination address
IR829(config-controller)#lte gps nmea ip udp 10.1.1.1 10.3.4.5 ?
    <0-65535> Destination port
IR829(config-controller)#lte gps nmea ip udp 10.1.1.1 10.3.4.5 3456
Cellular Modem in HWIC slot 0/0 is still in reset, we recommend to re-execute this cmd after
60 seconds
IR829(config-controller)#

```

The Command Line Interface for the gyroscope feature has been changed in IOS Release 15-7-3M1 in order to be compatible with the CCP Express NMS. The old CLI format was:

```

IR829(config)#gyroscope-reading frequency ?
1/min Reading 1 times per minute
1/sec Reading 1 time per second (default value)
10/min Reading 10 times per minute

```

From this release going forward, the format has been modified to:

```

IR829(config)#gyroscope-reading frequency ?
one/min Reading 1 times per minute
one/sec Reading 1 time per second (default value)
ten/min Reading 10 times per minute

```

After upgrading to this release, the router will have to be reconfigured.

GPS NMEA Multiple Stream

Feature is new for release 15.8(3)M1 and applies to the IR809 and IR829.

Previous versions of IOS only allowed for a GPS NMEA Stream for one device. This release has support for up to 6 devices at one time. The existing CLI **lte gps nmea ip udp** <src ip> <dest ip> <dest portno> under controller configuration has been enhanced.

Setting up the Configuration

To Enable GPS NMEA Multiple Stream:

```
Router# config t
Router(config)#controller cellular <Cellular Interface Number>
Router(config-controller)#lte gps nmea ip udp <source ip> <destination ip> <destination
port> stream <1-6>
```

To Disable GPS NMEA Multiple Stream:

```
Router(config-controller)#no lte gps nmea ip udp <source ip> <destination ip>
<destination port> stream <1-6>
```

Examples for Enabling/Disabling GPS NMEA Multiple Stream

Enable Example:

```
Router#(config-controller)#lte gps nmea ip udp 10.0.0.1 10.0.0.11 2020 ? stream GPS NMEA
multiple stream support
t
Router#(config-controller)#lte gps nmea ip udp 10.0.0.1 10.0.0.11 2020 stream ? <1-6> Stream
Number
Router#(config-controller)#lte gps nmea ip udp 10.0.0.1 10.0.0.11 2020 stream 6
```

Disable Example:

```
Router#(config-controller)#no lte gps nmea ip udp 10.0.0.1 10.0.0.11 2020 stream 6
```

GPS Multiple NMEA Stream Information

Use the show controller and show run configuration CLIs:

Sample Output

```
Router#sh cont cel 0 | inc NMEA
NMEA Stream no: 1 Configured
NMEA Stream no: 2 Configured
NMEA Stream no: 3 Not Configured
NMEA Stream no: 4 Configured
NMEA Stream no: 5 Configured
NMEA Stream no: 6 Not Configured
Router#sh run | sec cont
controller Cellular 0
```

```
lte gps nmea ip udp 10.10.0.1 10.10.0.10 2067 stream 1
lte gps nmea ip udp 20.20.0.1 20.25.0.20 2047 stream 2
lte gps nmea ip udp 20.27.0.1 20.27.0.20 2047 stream 4
lte gps nmea ip udp 20.20.0.1 20.20.0.20 2023 stream 5
```

Warning Messages

If the destination ip address and port number already exists:

```
Router#sh run | sec cont
controller Cellular 0
  lte gps mode standalone
  lte gps nmea ip udp 10.10.0.1 10.10.0.10 2067 stream 1
Router(config-controller)#lte gps nmea ip udp 10.10.0.1 10.10.0.10 2067 stream 5
  Destination ip address 10.10.0.10 and destination port number 2067 is already exists for
  the stream no:1.
```

Please use different destination ip address and port number.

If the stream number already exists:

```
Router#sh run | sec cont
controller Cellular 0
  lte gps mode standalone
  lte gps nmea ip udp 10.10.0.1 10.10.0.10 2067 stream 1
Router(config-controller)#lte gps nmea ip udp 20.20.0.1 20.20.0.10 2057 stream 1
  Stream number 1 is already active.
```

Please remove stream number configuration before creating it with different destination ip address and port number.

Troubleshooting the Cellular Interface

These procedures are to capture information to share with support in order to assist them in helping to troubleshoot an issue with the cellular interface. In order to capture logs, DM logs must be enabled. Refer to the following: <https://www.cisco.com/c/en/us/td/docs/routers/access/800/819/user/guide/3G4G-enhancements-userguide.html#pgfid-1063363>

The following are steps to capture Linux logs for the cellular interface.

Procedure

Step 1 Set up the fetch command.

Example:

```
# conf t
# service internal
# exit
# vds fetch-log
```

These steps will generate a directory on flash:vds-log.

Step 2 Capture the logs.

Example:

```

IR800# vds fetch-log
fetch: 4gmodem.log
      Sending file modes: C0644 510 4gmodem.log
fetch: auth.log
      Sending file modes: C0640 162330 auth.log
fetch: auth.log.1
      Sending file modes: C0640 262215 auth.log.1
fetch: auth.log.2.gz
      Sending file modes: C0640 11297 auth.log.2.gz
fetch: auth.log.3.gz
      Sending file modes: C0640 11296 auth.log.3.gz
fetch: cwan_modem0.log
      Sending file modes: C0644 3875716 cwan_modem0.log
fetch: cwan_modem1.log
      Sending file modes: C0644 791629 cwan_modem1.log
fetch: daemon.log
      Sending file modes: C0640 1404 daemon.log
fetch: dmesg
      Sending file modes: C0644 13740 dmesg
fetch: dmesg.0
      Sending file modes: C0644 0 dmesg.0
fetch: ios_cs_verify.log
      Sending file modes: C0644 1091 ios_cs_verify.log
fetch: ios_vds_com.log
      Sending file modes: C0644 219169 ios_vds_com.log
fetch: ios_vds_com.log.1
      Sending file modes: C0644 262207 ios_vds_com.log.1
fetch: ios_vds_com.log.2.gz
      Sending file modes: C0644 7859 ios_vds_com.log.2.gz
fetch: ios_vds_com.log.3.gz
      Sending file modes: C0644 7894 ios_vds_com.log.3.gz
fetch: kern.log
      Sending file modes: C0640 38608 kern.log
fetch: messages
      Sending file modes: C0640 174064 messages
fetch: messages.1
      Sending file modes: C0640 262364 messages.1
fetch: messages.2.gz
      Sending file modes: C0640 18434 messages.2.gz
fetch: messages.3.gz
      Sending file modes: C0640 25027 messages.3.gz
fetch: udev
      Sending file modes: C0644 124266 udev
fetch: vdscli-acpid.log
      Send

```

Step 3 Stop the logging after 10 minutes.

Step 4 View the flash directory, and you will see the vds-log directory.

Example:

```

IR800# dir flash:
Directory of flash:/
16 -rw-          660 Nov 11 2016 19:25:20 +00:00  vlan.dat
1  drw-           0 Jan 1 2014 16:27:44 +00:00  7455_02.18.02.00_Verizon_002.022_000
17 -rw-    160368465 Nov 11 2016 19:35:30 +00:00  ir800-universalk9-bundle.SPA.156-3.M0a
18 -rw-     63753008 Nov 11 2016 19:45:34 +00:00  ir800-universalk9-mz.SPA.156-3.M0a
19 -rw-     64381598 Nov 11 2016 19:50:24 +00:00  74XX_02.20.03.00.cwe

```

```

20 -rw-          9143 Nov 11 2016 19:59:30 +00:00 7455_02.20.03.00_ATT_002.019_000.nvu
4  drw-          0 Jan 1 2014 16:17:58 +00:00 managed
14 drw-          0 Jan 1 2014 16:17:58 +00:00 eem
15 -rw- 62582707 Jan 1 2014 16:27:24 +00:00 ir800-universalk9-mz.SSA.156-20160701_225522
21 -rw- 161162048 Nov 16 2016 18:41:46 +00:00 ir800-universalk9-bundle.SSA.156-2.10.62.GB
22 -rw- 63827874 Nov 16 2016 18:54:30 +00:00 ir800-universalk9-mz.SSA.156-2.10.62.GB
23 drw-          0 Nov 16 2016 19:06:34 +00:00 vds-log

```

Step 5 The flash:/vds-log directory contains the log files captured.

Example:

```

24 -rw-          510 Nov 16 2016 19:06:44 +00:00 4gmodem.log
25 -rw-       162330 Nov 16 2016 19:06:54 +00:00 auth.log
26 -rw-       262215 Nov 16 2016 19:07:04 +00:00 auth.log.1
27 -rw-       11297 Nov 16 2016 19:07:16 +00:00 auth.log.2.gz
28 -rw-       11296 Nov 16 2016 19:07:24 +00:00 auth.log.3.gz
29 -rw-     3875716 Nov 16 2016 19:07:42 +00:00 cwan_modem0.log
30 -rw-       791629 Nov 16 2016 19:07:54 +00:00 cwan_modem1.log
31 -rw-        1404 Nov 16 2016 19:08:04 +00:00 daemon.log
32 -rw-       13740 Nov 16 2016 19:08:14 +00:00 dmesg
33 -rw-          0 Nov 16 2016 19:08:24 +00:00 dmesg.0
34 -rw-        1091 Nov 16 2016 19:08:32 +00:00 ios_cs_verify.log
35 -rw-       219169 Nov 16 2016 19:08:42 +00:00 ios_vds_com.log
36 -rw-       262207 Nov 16 2016 19:08:54 +00:00 ios_vds_com.log.1
37 -rw-        7859 Nov 16 2016 19:09:04 +00:00 ios_vds_com.log.2.gz
38 -rw-        7894 Nov 16 2016 19:09:14 +00:00 ios_vds_com.log.3.gz
39 -rw-       38608 Nov 16 2016 19:09:24 +00:00 kern.log
40 -rw-      174064 Nov 16 2016 19:09:34 +00:00 messages
41 -rw-       262364 Nov 16 2016 19:09:44 +00:00 messages.1
42 -rw-       18434 Nov 16 2016 19:09:54 +00:00 messages.2.gz
43 -rw-       25027 Nov 16 2016 19:10:04 +00:00 messages.3.gz
44 -rw-      124266 Nov 16 2016 19:10:14 +00:00 udev
45 -rw-         292 Nov 16 2016 19:10:24 +00:00 vdscli-acpid.log
46 -rw-         909 Nov 16 2016 19:10:34 +00:00 vdscli-eventd.log
47 -rw-         467 Nov 16 2016 19:10:44 +00:00 vdscli-vdscli-bde-gos.log
48 -rw-         479 Nov 16 2016 19:10:54 +00:00 vdscli-vdscli-bde-ir800.log
49 -rw-          81 Nov 16 2016 19:11:04 +00:00 vdscli-wiredd.log
50 -rw-      140382 Nov 16 2016 19:11:14 +00:00 vdscli-wirelessd.log
51 -rw-         1192 Nov 16 2016 19:11:24 +00:00 vdscli.log
994918400 bytes total (34735718)

```

What to do next

Other command output that will be helpful to collect for your business unit contact:

```

# Show platform hypervisor
# Show platform led
# Show tech
# Show cellular 0/0 all
# Show controller 0/0
# Show interface cellular 0/0
# Show ip interface brief
# Show running-config

```

