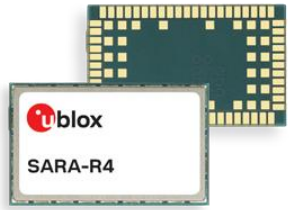
SARA-R410M/SARA-R412M

AT command connect to AWS IoT core

Application note



Abstract

This document provides examples of how to use AT commands to connect the AWS IoT service with u-blox SARA-R410M / SAR-R412M.

UBX-20010011 - R02
C1-Public

Document information

Title	**SARA-R410M/SARA-R412M**	
Subtitle	AT command connect to AWS IoT core	
Document type	Application note	
Document number	UBX-20010011	
Revision and date	R02	29-Mar-2021
Disclosure restriction	C1-Public	

Product status	**Corresponding content status**	
Functional sample	Draft	For functional testing. Revised and supplementary data will be published later.
In development / Prototype	Objective specification	Target values. Revised and supplementary data will be published later.
Engineering sample	Advance information	Data based on early testing. Revised and supplementary data will be published later.
Initial production	Early production information	Data from product verification. Revised and supplementary data may be published later.
Mass production / End of life	Production information	Document contains the final product specification.

This document applies to the following products:

Product name
SARA-R410M
SARA-R412M

Contents

1 Steps for getting started with AWS IoT

To get started with AWS IoT service, follow the steps shown on the AWS website:

https://docs.aws.amazon.com/iot/latest/developerguide/iot-gs.html

You can also get an AWS IoT certification, though currently only a legacy certification is supported:

https://docs.aws.amazon.com/iot/latest/developerguide/create-device-certificate.html

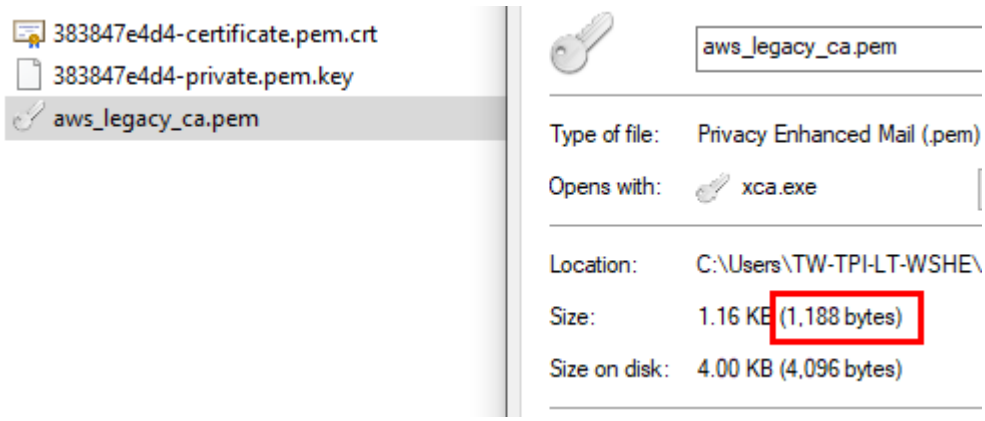
For more details on AT commands, see SARA-R4 AT commands manual [2].

⚠ Due to AWS's continuous evolution, some information provided in this document can be not up to date.

1.1 Store certifications in module flash

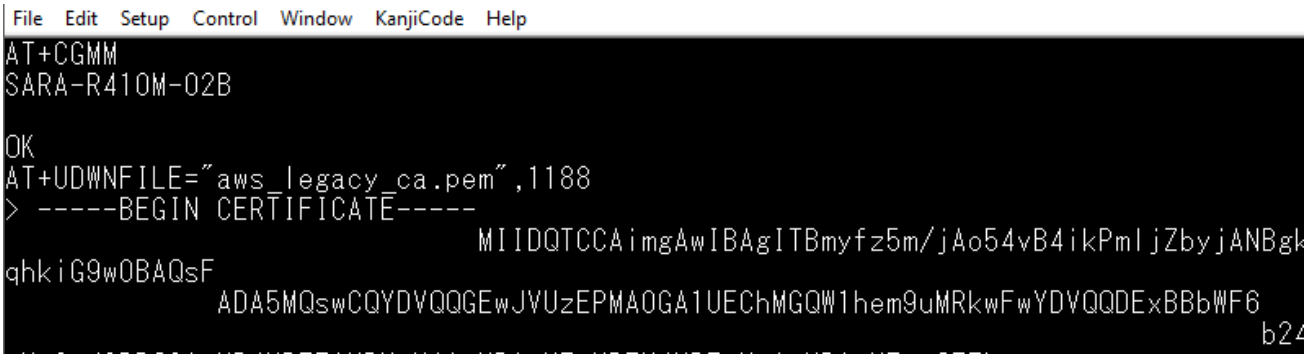
After downloading the CA, CC, and PK from AWS, store them in the module via AT commands. Here are the steps to download files to the module's flash memory:

1.1.1 Check the file size



1.1.2 Use terminal software to write the file in the module

In the following example TeraTerm is used to write CA, CC, and PK in the module. After character ">" choose File tab->Send file-> Select "aws_legacy_ca.pem"



1.1.3 File stored successfully

```
            5MsI+yMRQ+hDKXJioaIdXgjUkK642M4UwtBV8ob2xJNDd2ZhwLnoQdeXeGADbkpy
                                                                        rqX
RfboQnoZsG4q5WTP468SQvvG5
                        -----END CERTIFICATE-----


OK
```

1.1.4 Stored the 3 files in the module flash

Repeat steps 1.1.1 - 1.1.3 to download the other files "383847e4d4-certificate.pem.crt" and "383847e4d4-private.pem.key".

1.2 Check CA, CC, and PK in file system

Command	Response	Description
AT+ULSTFILE=2,"aws_legacy_ca.pem"	+ULSTFILE: 1188 OK	CA availability in the module.
AT+ULSTFILE=2,"383847e4d4-certificate.pem.crt"	+ULSTFILE: 1224 OK	CC availability in the module.
AT+ULSTFILE=2,"383847e4d4-private.pem.key"	+ULSTFILE: 1679 OK	PK availability in the module

1.3 Import CA, CC, and PK from a file store on file system

Command	Response	Description
AT+USECMNG=1,0,"aws_legacy_ca.pem","aws_legacy_ca.pem"	+USECMNG: 1,0,"aws_legacy_ca.pem","CB17E431673EE209FE455793F30AFA1C" OK	Import CA.
AT+USECMNG=1,1,"383847e4d4-certificate.pem.crt","383847e4d4-certificate.pem.crt"	+USECMNG: 1,1,"383847e4d4-certificate.pem.crt","50C3004AAE690124E3D7F96F904D7084" OK	Import CC.
AT+USECMNG=1,2,"383847e4d4-private.pem.key","383847e4d4-private.pem.key"	+USECMNG: 1,2,"383847e4d4-private.pem.key","CD879AA22744A7211D3AF5D3BEFAFF29" OK	Import PK.

1.4 Enable HEX mode and set security profile

Command	Response	Description
AT+UDCONF=1,1	OK	Enable the HEX mode.
AT+USECPRF=0,0,1	OK	Set the certificate validation level 1.
AT+USECPRF=0,1,0	OK	Set the TLS version to any.
AT+USECPRF=0,2,0	OK	Set automatic the cipher suite.
AT+USECPRF=0,3,"aws_legacy_ca.pem"	OK	Set the trusted root certificate internal name.
AT+USECPRF=0,5,"383847e4d4-certificate.pem.crt"	OK	Set the client certificate internal name.
AT+USECPRF=0,6,"383847e4d4-private.pem.key"	OK	Set the client certificate internal name.

Command	Response	Description
AT+USECPRF=0,10,"[redacted].iot.ap-northeast-1.amazonaws.com "	OK	Set the Server Name Indication. ☞ SNI is a feature of SSL/TLS which uses an additional SSL/TLS extension header to specify the server name to which the client is connecting to. SNI configuration may be required to support the certificate handling used with virtual hosting provided by the various SSL/TLS enabled servers mostly in cloud-based infrastructures.

1.5 Create TCP socket and connect to AWS IoT with SSL enable

Use the +COPS read command to check the network registrations status.

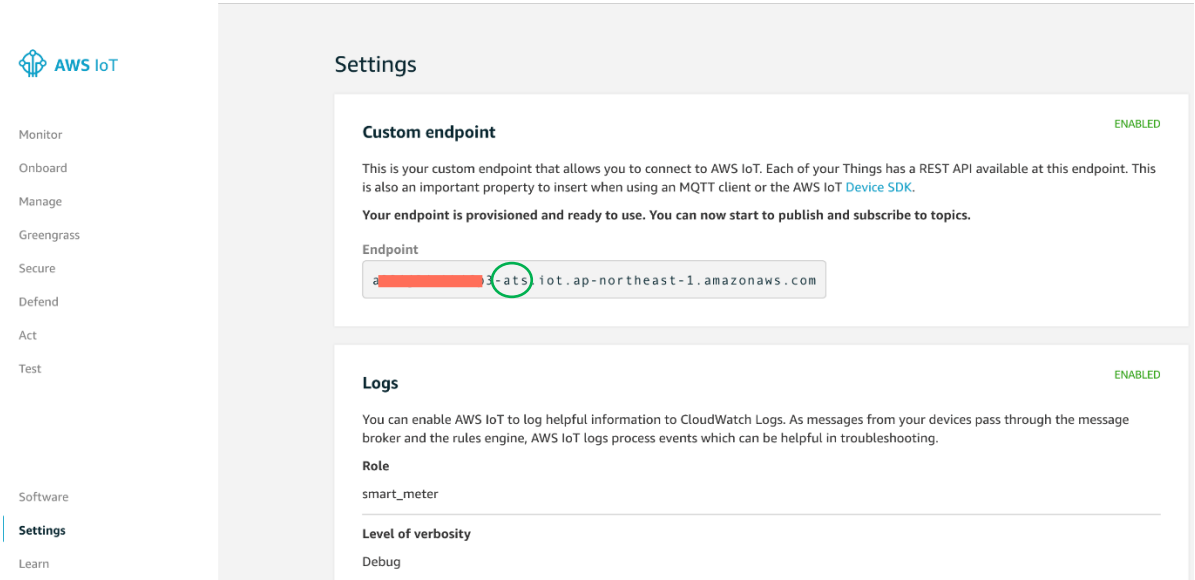
After the device has been registered to the network, create a TCP socket to connect with.

To get AWS end point, follow the steps on the website:

Command	Response	Description
AT+USOCR=6	+USOCR: 0 OK	Create TCP socket.
AT+USOSEC=0,1,0	OK	Enable SSL/TLS connection on a TCP socket.
AT+USOCO=0,"[redacted].iot.ap-northeast-1.amazonaws.com",8883	OK	Connect to AWS IoT server by AT command.

☞ To get the end point, it should be on AWS account > Settings > Endpoint. It should delete **"-ats"** because currently only legacy certification can be supported.

☞ AWS IoT Core is currently supported using the legacy root CA certificate in a limited number of AWS regions. For the list of supported AWS region visit the following page: https://docs.aws.amazon.com/general/latest/gr/greengrass.html#greengrass-legacy-endpoints



2 Send MQTT message from module to AWS IoT core

MQTT messages require conversion from ASCII to hexadecimal format. The arguments for these messages include the MQTT topic and payload. The messages have been created by the AWS IoT SDK. For more details, see the website for AWS IoT SDKs:

https://docs.aws.amazon.com/iot/latest/developerguide/iot-sdks.html

The examples here are using Python.

Connect the end point with default connection header, Client ID, and protocol.

ASCII message	MQTT_Test ?SDK=Python&Version=1.4.7
HEX number	103000044d5154540482025800094d5154545f5465737400193f53444b3d507974686f6e2656657273696f6e3d312e342e37
AT command	AT+USOWR=0,50,"103000044d5154540482025800094d5154545f5465737400193f53444b3d507974686f6e2656657273696f6e3d312e342e37"

2.1 Subscribe to a topic and receive a message from AWS IoT core

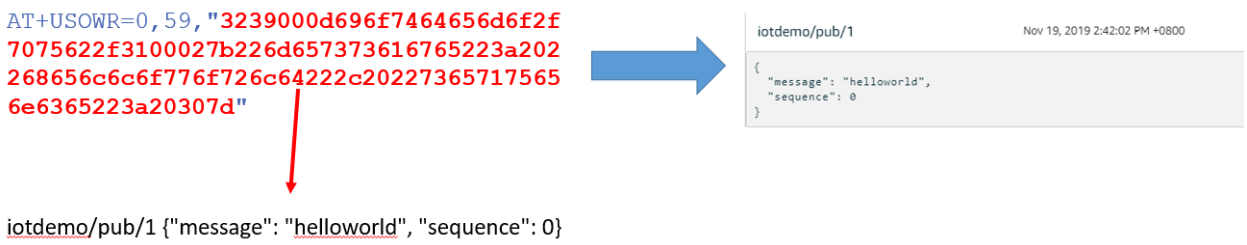
Subscribe topic: iotdemo/pub/1

ASCII message	iotdemo/pub/1
HEX number	82120001000d696f7464656d6f2f7075622f3101
AT command	AT+USOWR=0,20,"82120001000d696f7464656d6f2f7075622f3101”

2.2 Publish message to AWS IoT core

Publish message: iotdemo/pub/1{"message": "helloworld", "sequence": 0}

ASCII message	iotdemo/pub/1{"message": "helloworld", "sequence": 0}
HEX number	3239000d696f7464656d6f2f7075622f3100027b226d657373616765223a202268656c6c6f776f726c64222c202273657175656e6365223a20307d
AT command	AT+USOWR=0,59,"3239000d696f7464656d6f2f7075622f3100027b226d657373616765223a202268656c6c6f776f726c64222c202273657175656e6365223a20307d"



☞ For more details about the conversion from ASCII to HEX format, see appendix A.

3 Using AWS IoT device shadow

When AWS IoT Core registers a thing, a shadow can be used to interact with the device. For more details, see:

https://docs.aws.amazon.com/iot/latest/developerguide/device-shadow-data-flow.html

Example: When you register "ublox_sara_r401m" as a thing, then its reversed MQTT topic for shadow would be:

MQTT

Use topics to enable applications and things to get, update, or delete the state information for a Thing (Thing Shadow)

Learn more

Update to this thing shadow

```
$aws/things/ublox_sara_r401m/shadow/update
```

Update to this thing shadow was accepted

```
$aws/things/ublox_sara_r401m/shadow/update/accepted
```

Update this thing shadow documents

```
$aws/things/ublox_sara_r401m/shadow/update/documents
```

Update to this thing shadow was rejected

```
$aws/things/ublox_sara_r401m/shadow/update/rejected
```

Get this thing shadow

```
$aws/things/ublox_sara_r401m/shadow/get
```

Get this thing shadow accepted

```
$aws/things/ublox_sara_r401m/shadow/get/accepted
```

Getting this thing shadow was rejected

```
$aws/things/ublox_sara_r401m/shadow/get/rejected
```

Delete this thing shadow

```
$aws/things/ublox_sara_r401m/shadow/delete
```

Deleting this thing shadow was accepted

```
$aws/things/ublox_sara_r401m/shadow/delete/accepted
```

Deleting this thing shadow was rejected

```
$aws/things/ublox_sara_r401m/shadow/delete/rejected
```

3.1 Update the contents of a device shadow

Boot up the device and issue the +USOWR AT command to publish updates to shadow service from the u-blox cellular module.

```
AT+USOWR=0,141,"308a01002a246177732f7468696e67732f7
5626c6f785f736172615f723430316d2f736861646f772f7570
646174657b227374617465223a207b226465736972656422 3a2
07b2270726f7065727479223a20307d7d2c2022636c69656e74
546f6b656e223a202262353264323664612d313464302d34333
6612d383764382d3662393333306362623137322 27d"

+USOWR: 0,141

OK
```

$aws/things/ublox_sara_r401m/shadow/update{"state": {"desired": {"property": 0}}, "clientToken": "b52d26da-14d0-436a-87d8-6b9330cbb172"}

$aws/things/ublox_sara_r401m/shadow/updat... Nov 19, 2019 5:12:26 PM +0800


```
{
  "state": {
    "desired": {
      "property": 0
    }
  },
  "metadata": {
    "desired": {
      "property": {
        "timestamp": 1574154746
      }
    }
  },
  "version": 68,
  "timestamp": 1574154746,
  "clientToken": "b52d26da-14d0-436a-87d8-6b9330cbb172"
}
```


3.2 Subscribe and retrieve the latest state stored in device shadow

Boot up the device and issue the +USOWR AT command to subscribe to a shadow topic from the shadow service, and then use “AT+USORD” to receive subscribed shadow message.

```
AT+USOWR=0,55,"82350001003024617773 2f746869
6e67732f75626c6f785f736172615f723430316d2f7
36861646f772f7570646174652f64656c746100"
```

Subscribe topic: $aws/things/ublox_sara_r401m/shadow/update/delta

```
AT+USORD=0,100
+USORD:
0,100,"30D4010030246177732F7468696E67732F75626C6F785
F736172615F723430316D2F736861646F772F7570646174652F
64656C74617B2276657273696F6E223A37332C2274696D6573
74616D70223A313537343135353637332C227374617465223A
7B22"
OK
```

equal

Received subscribed message:
$aws/things/ublox_sara_r401m/shadow/update{"state": {"desired": {"property": 0}}, "clientToken": "b52d26da-14d0-436a-87d8-6b9330cbb172"}

$aws/things/ublox_sara_r401m/shadow/updat... Nov 19, 2019 5:27:53 PM +0800


```
{
  "state": {
    "desired": {
      "property": 0
    }
  },
  "metadata": {
    "desired": {
      "property": {
        "timestamp": 1574155673
      }
    }
  },
  "version": 73,
  "timestamp": 1574155673,
  "clientToken": "b52d26da-14d0-436a-87d8-6b9330cbb172"
}
```


☞ As described in section 2, convert ASCII to HEX. See appendix A for information about how to convert from ASCII to HEX.

Appendix

A How to convert ASCII to HEX

You can use this website tool to convert ASCII to HEX:

https://www.rapidtables.com/convert/number/ascii-to-hex.html

Related documentation

[1] u-blox SARA-R4 series data sheet, UBX-16024152
[2] u-blox SARA-R4 series AT commands manual, UBX-17003787
[3] u-blox SARA-R4 series system integration manual, UBX-16029218

☞ For regular updates to u-blox documentation and to receive product change notifications, register on our homepage (www.u-blox.com).

Revision history

Revision	Date	Name	Comments
R01	12-Mar-2020	wshe	Initial release
R02	29-Mar-2021	alos	Generic formal improvements

Contact

For complete contact information, visit us at www.u-blox.com.

u-blox Offices

North, Central and South America

u-blox America, Inc.

Phone: +1 703 483 3180
E-mail: info_us@u-blox.com

Regional Office West Coast:

Phone: +1 408 573 3640
E-mail: info_us@u-blox.com

Technical Support:

Phone: +1 703 483 3185
E-mail: support@u-blox.com

Headquarters
Europe, Middle East, Africa

u-blox AG

Phone: +41 44 722 74 44
E-mail: info@u-blox.com
Support: support@u-blox.com

Asia, Australia, Pacific

u-blox Singapore Pte. Ltd.

Phone: +65 6734 3811
E-mail: info_ap@u-blox.com
Support: support_ap@u-blox.com

Regional Office Australia:

Phone: +61 3 9566 7255
E-mail: info_anz@u-blox.com
Support: support_ap@u-blox.com

Regional Office China (Beijing):

Phone: +86 10 68 133 545
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office China (Chongqing):

Phone: +86 23 6815 1588
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office China (Shanghai):

Phone: +86 21 6090 4832
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office China (Shenzhen):

Phone: +86 755 8627 1083
E-mail: info_cn@u-blox.com
Support: support_cn@u-blox.com

Regional Office India:

Phone: +91 80 405 092 00
E-mail: info_in@u-blox.com
Support: support_in@u-blox.com

Regional Office Japan (Osaka):

Phone: +81 6 6941 3660
E-mail: info_jp@u-blox.com
Support: support_jp@u-blox.com

Regional Office Japan (Tokyo):

Phone: +81 3 5775 3850
E-mail: info_jp@u-blox.com
Support: support_jp@u-blox.com

Regional Office Korea:

Phone: +82 2 542 0861
E-mail: info_kr@u-blox.com
Support: support_kr@u-blox.com

Regional Office Taiwan:

Phone: +886 2 2657 1090
E-mail: info_tw@u-blox.com
Support: support_tw@u-blox.com