AWS IoT AT Command Reference For MikroElektronika AnyNet 2G click

MikroElektronika DEVELOPMENT TOOLS | COMPILERS | BOOKS	www.mikroe.com	MikroElektronika D.O.O. Batajnički drum 23 11186 Zemun, Belgrade Serbia Office: + 381 11 78 57 600 Support: + 381 11 78 57 628
eseye INTELLIGENTLY CONNECTED AnyNet Secure	www.eseye.com	Eseye Limited 8 Frederick Sanger Road Surrey Research Park Guildford Surrey, GU2 7YD United Kingdom

Table of Contents

Introduction

AnyNet 2G click module has been designed to give the easiest method of connecting IoT Devices into the AWS IoT Platform. This document describes available AT commands on the Serial interface.

AnyNet 2G click module includes an Eseye AnyNet Secure SIM card, that provides connectivity in over 75 countries (billed through your AWS account).

For more information on how the AnyNet Secure Cellular Connectivity combines with the AWS IoT cloud see: https://www.youtube.com/watch?v=bDXk-yCoxxo

To use the AnyNet 2G click module and the commands in this user guide, you require:

An active AWS account – for help see: https://aws.amazon.com/premiumsupport/knowledge-center/create-and-activate-aws-account/

The Eseye AWS Marketplace Integration from: https://aws.amazon.com/marketplace/pp/B073S37V78

Module Overview

AT Command Interface

The Serial interface to the Click board™ offers custom AT commands to allow data to be sent to the AWS IoT Platform, and extracted from it.

AnyNet Secure Firmware

The Eseye provided embedded firmware runs on the Click board™. This firmware handles loading the security material from the AnyNet SIM and establishing the MQTT / TLS links. Additionally, the firmware handles the link maintenance based upon the application requirements.

AnyNet Secure SIM

The Eseye AnyNet Secure SIM is soldered on the AnyNet 2G click. It combines a multi profile global SIM solution (AnyNet) and the AWS security provisioning functionality.

Serial Interface

The Serial Interface operates at 9600 baud, 8n1

Power requirements and other interfaces

Refer to the MikroElektronika documentation https://www.mikroe.com/anynet-2g-click

AnyNet Click AWS AT Commands

These are the AT command extensions that will be handled by the AnyNet Secure firmware to support easy communication with AWS IoT.

General Commands

AT+QCCID	Query ICCID
Read Command **AT+QCCID**	Response **ICCID of the SIM** **OK**

AT+GSN	Query IMEI
Read Command **AT+GSN**	Response **IMEI of the modem on the board** **OK**

AT+AWSVER	Query Version
Read Command **AT+AWSVER**	Response **VERSION info for the AnyNet AWS IoT code** **OK**

AT+AWSRESET	Force a reload of parameters from the SIM card
Read Command **AT+AWSRESET**	Response **OK** The AnyNet Click will restart at this point

AT+AWSSTATE	Check AWS State
Test Command **AT+AWSSTATE=?**	Response **+AWSSTATE: (0-8)** **OK**
Read Command **AT+AWSSTATE?**	Response **+AWSSTATE: <state>** **OK**

Parameter	
<state>	AWS system state 1 Idle 2 Waiting Keys 3 Connecting to network 4 Establishing SSL 5 SSL Connected 6 Connecting MQTT 7 Ready: MQTT Connected 8 Ready: Subscribed 9 Error

Operation

The act of opening a topic for publishing or subscribing will cause the modem to read the security information from the SIM card and establish a connection to the AWS IoT service.

AWS IoT Button Commands

AT+AWSBUTTON	Send an IoT Button Press Message
Write Command **AT+AWSBUTTON=<click type>**	Response If the format is correct the response will be **OK** Otherwise the response will be **ERROR** **The following URC will notify the result of message delivery** **+SEND OK**

Parameter

<click type>	SHORT LONG DOUBLE

The topic to which the message is published will be combined with the iotbutton/<IMEI>/

The message will contain the button press type, and the analogue voltage measured on the module analogue input pin.[1]

1 See https://www.mikroe.com/anynet-2g-click

Publishing to IoT Service commands

AT+AWSPUBOPEN	Open AWS Publish Topic
Test Command **AT+AWSPUBOPEN=?**	Response **+AWSPUBOPEN: (0-1),<awstopic>** **OK**
Read Command **AT+AWSPUBOPEN?**	Response list of configured publish topics in the format **<awspindex>: <awstopic>/<thingname>** **OK**
Write Command[2] **AT+AWSPUBOPEN=<awspindex>, "<awstopic>"**	Response If the format is correct the response will be **OK** Otherwise the response will be **ERROR** **The following URC will notify the result of the connection.** **+AWSPUBOPEN: <awspindex>,<aswconcode>**

Parameter

<awspindex>	AWS publish socket index 0-1
<awstopic>	String Type, AWS publish topic path
<awsconcode>	1 Success -1 Error -2 Socket Already in use

<awstopic> will be combined with the “Thing Name” that is delivered to the Anynet secure SIM during device provisioning to form the complete subscribed resource name in the form <awstopic>/<thing_name>

e.g. temperature/thing_0001

The ‘/’ and Thing name do not need to be present in the AWSPUBOPEN command as these will be added to the subscription automatically.

AT+AWSPUBLISH	AWS Publish
Test Command **AT+AWSPUBLISH=?**	Response **AWSPUBLISH: (0-1), (1-240)[,(0-1)]**

2 Note that the aws topic must be contained within quotes

	OK
Read Command **AT_AWSPUBLISH?**	Response **OK**
Write Command **AT+AWSPUBLISH=<awspindex>,<length>[,<qos>]**	Response If the topic connection is not open, if the device is unable to accept data or some other error occurs: **ERROR** Response **>** At the prompt input the data to be sent on this topic, data is accepted for the topic until <length> bytes are input If sending is successful: **SEND OK** If the process of sending data is blocked: **SEND FAIL**

Parameter	
<awspindex>	AWS publish socket index 0-1
<length>	Length of data to be delivered for this topic
<qos>	The QOS requirements on the delivery of this data. 2 Send, no acknowledgement required. SEND OK returns immediately 3 Send, acknowledgement required. SEND OK return on acknowledgement If not specified the default QOS is 1

Commands to subscribe to IoT topics

AT+AWSSUBOPEN	Open AWS Subscribe Topic
Test Command **AT+AWSSUBOPEN=?**	Response **+AWSSUBOPEN: (0-1),<awstopic>** **OK**
Read Command **AT+AWSSUBOPEN?**	Response list of subscriber topics in the format **<awssindex>: <awstopic>/<thingname>** **OK**
Write Command[3] **AT+AWSSUBOPEN=<awssindex>, "<awstopic>"**	Response If the format is correct the response will be **OK** Otherwise the response will be **ERROR** **The following URC will notify the result of the connection.** **+AWSSUBOPEN: <awssindex>,<aswconcode>**
Parameter **Parameter**	
<awssindex>	AWS subscribed socket index 0-1
<awstopic>	String Type, AWS publish topic path
<awsconcode>	1 Success -1 Error -2 Socket Already in use

<awstopic> will be combined with the “Thing Name” that is delivered to the Anynet secure SIM during device provisioning to form the complete subscribed resource name in the form <awstopic>/<thing_name>

e.g. resolution/thing_0001

The ‘/’ and Thing name do not need to be present in the AWSSUBOPEN command as these will be added to the subscription automatically.

3 Note that the aws topic must be contained within quotes

AWS Notify of Subscribed Data	
+AWS:<awssindex>,<length> <CR><LF><Binary Data>	
Reference	This URC is to present subscribed data for a topic

Parameter

<awssindex>	AWS subscribed socket index 0-1
<length>	Length of data to be delivered for this topic

AT+AWSSUBCLOSE	Close AWS Subscribed Topic
Test Command **AT+AWSSUBCLOSE=?**	Response **+AWSSUBCLOSE: (0-7)** **OK**
Read Command **AT+AWSSUBCLOSE?**	Response **OK**
Write Command **AT+AWSSUBCLOSE=<awspindex>**	Response If the format is correct the response will be **OK** Otherwise the response will be **ERROR** **The following URC will notify the result of the connection.** **+AWSPUBCLOSE: <awspindex>, <aswconcode>**

Parameter	
<awspindex>	AWS subscribed socket index 0-7
<awsconcode>	1 Success -1 Error -2 Socket Not Open