

Getting started guide

Functionality

The R718G is a Wireless Light Sensor for Netvox ClassA type devices based on the LoRaWAN open protocol and is compatible with the LoRaWAN protocol. It can detect illuminance anywhere, ex. sunlight, outdoor area.

Install CLI for AWS IoT Things Graph

```
## Install AWS CLI
```

```
https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-install.html
```

```
## Install preview Things Graph API models
```

```
https://docs.aws.amazon.com/cli/latest/reference/configure/add-model.html
```

```
aws configure add-model --service-name iotthingsgraph --service-model file://service-2.json
```

```
## Install jq
```

```
https://stedolan.github.io/jq/
```

```
## Verify preview model installed
```

```
aws iotthingsgraph map-property help
```

```
expected to see help output instead of "Invalid choice" error message
```

Find your desired device

1. Go to AWS device catalog page: <https://devices.amazonaws.com>
2. Locate your desired sensor.
3. For private beta, Things Graph is using API v1. For API v1, use the model file sensor_model.json (device manufacturer enters their model file name). For production, Things Graph will use API v2. For API v2, please use the S3 URL link that you see in the Device catalog page.

Using the device model

The following tests will show the different use cases and binary messages than be used with the model. You can use the binary test data to verify correct functionality of the model.

1. Test name: Report Data Command
 - a. Mapping used: map_R718G_Uplink
 - b. Use case tested: Report Data
 - c. Payload format: port = 6

Bytes	1	1	1	Var(Fix=8 Bytes)
	Version	DeviceType	ReportType	NetvoxPayloadData

Version: 1 bytes – 0x01—the Version of NetvoxLoRaWAN Application Command Version

DeviceType: 1 byte – Device Type of Device

ReportType: 1 byte—the Presentation of the NetvoxPayloadData, according the devicetype

NetvoxPayloadData: Fixed bytes (Fixed = 8bytes)

Device	DeviceType	ReportType	NetvoxPayloadData		
R718G	0x1E	0x01	Battery (1Byte, unit:0.1V)	illuminance (4 Bytes, unit: 1 Lux)	Reserved (3 Bytes, Fixed 0x00)

- d. Input binary data: 011E0124000000A0000000
 (DeviceType: 0x1E, ReportType: 0x01, Battery: 24_{hex} = 36_{dec},
 illuminance: 000000A0_{hex} = 160_{dec})

- e. API call (V1):

```
aws iotthingsgraph map-property \
  --region us-east-1 \
  --endpoint-url "https://iotthingsgraph.us-east-1.amazonaws.com" \
  --namespace-snapshot file:///path/to/model/json/file \
  --property-value "011E0124000000A0000000" \
  --mapping-info '{"mappingId":"urn:tdm:us-east-1/000000000000/default:mapping:map_R718G_Uplink","mappingDirection": "FORWARD",
  "contextProvider":{"json": "{\"port\":6}"}}'
```

- f. Expected results (V1)

```
{
  "propertyId": "Netvox_R718G_Uplink/types/UplinkPort6Payload",
  "propertyValue": "{
    \"DeviceType\": \"R718G\",
    \"Version\": 1,
    \"Battery\": 3.6,
    \"ReportType\": 1,
    \"Illuminance\": 160
  }"
}
```

2. Test name: Set and read report configuration
 - a. Mapping used: map_R718G_Uplink
 - b. Use case tested: Configure and read report parameters
 - c. Payload format: port = 7

Bytes	1	1	Var(Fix =9 Bytes)
	CmdID	DeviceType	NetvoxPayloadData

CmdID– 1 bytes

DeviceType– 1 byte – Device Type of Device

NetvoxPayloadData– var bytes (Max=9 bytes)

Description	CmdID	Device Type	NetvoxPayloadData			
Config ReportReq	0x01	0x1E	MinTime (2bytes Unit:s)	MaxTime (2bytes Unit:s)	Battery Change (1byte Unit:0.1v)	IlluminanceChange (4 Bytes, Unit:1 Lux)
Config ReportRsp	0x81		Status (0x00:success)	Reserved (8 Bytes, Fixed 0x00)		
ReadConfig ReportReq	0x02		Reserved (9 Bytes, Fixed 0x00)			
ReadConfig ReportRsp	0x82		MinTime (2bytes Unit:s)	MaxTime (2bytes Unit:s)	Battery Change (1byte Unit:0.1v)	IlluminanceChange (4 Bytes, Unit:1 Lux)
SetSunlight SampleRange Req	0x03		RangeSetting(1 Byte 0x00: 3 ~ 65000 lux, 0x01: 3 ~ 130000 lux, 0x02: 3 ~ 220000 lux)	Reserved (8 Bytes, Fixed 0x00)		
SetSunlight SampleRange Rsp	0x83		Status (0x00:success)	Reserved (8 Bytes, Fixed 0x00)		
GetSunlight SampleRange Req	0x04		Reserved (9 Bytes, Fixed 0x00)			
GetSunlight SampleRange Rsp	0x84		RangeSetting(1 Byte 0x00: 3 ~ 65000 lux, 0x01: 3 ~ 130000 lux, 0x02: 3 ~ 220000 lux)	Reserved (8 Bytes, Fixed 0x00)		

d. Input binary data:

i. **Set Report Configuration**

MinTime = 5min(300s), MaxTime = 15min(900s), BatteryChange = 0.1v,
IlluminanceChange = 100 Lux

Downlink: 011E012C03840100000064 012C_{hex} = 300_{dec}, 0384_{hex} = 900_{dec},
0.1v(Unit:0.1v) => 0.1 ÷ 0.1 = 1, 01_{hex} = 1_{dec},
IlluminanceChange:00000064_{hex} = 100_{dec}

Response: 811E00000000000000000000 (Configuration success)
811E01000000000000000000 (Configuration failure)

ii. **Read Report Configuration**

Downlink: 021E00000000000000000000
Response: 821E012C03840100000064 (Current configuration)

iii. **Set Sunlight Sample Range**

For example, SunlightSampleRange = 0x02 (3-220000 Lux)

Downlink: 031E02000000000000000000
Response: 831E00000000000000000000 (Configuration success)
831E01000000000000000000 (Configuration failure)

iv. **Get Sunlight Sample Range**

Downlink: 041E00000000000000000000
Response: 841E02000000000000000000 (Current configuration)

e. API call (Set report configuration):

```
aws iotthingsgraph map-property \  
  --region us-east-1 \  
  --endpoint-url "https://iotthingsgraph.us-east-1.amazonaws.com" \  
  --namespace-snapshot file:///path/to/model/json/file \  
  --property-value "011E012C03840100000064" \  
  --mapping-info '{"mappingId":"urn:tdm:us-east-1/000000000000/default:mapping:map_R718G_Uplink","mappingDirection": "FORWARD",  
"contextProvider":{"json": "{\\"port\\":7}"}}'
```

f. Expected results

```
{  
  "propertyId":{"  
    "DeviceType":"Netvox_R718G_Uplink/types/DeviceTypeEnum",  
    "CmdId":"Netvox_R718G_Uplink/types/CmdIdEnum",  
    "Payload":"Netvox_R718G_Uplink/types/ConfigureCmdPayload"  
  }},  
  "propertyValue":{"  
    "DeviceType":"R718G",  
    "CmdId":"ConfigReportReq",  
    "Payload":{"  
      "IlluminanceChange":100,  
      "MaxTime":900,  
      "MinTime":300,  
      "BatteryChange":0.1  
    }  
  }  
}
```

g. Expected results (input data: 811E000000000000000000)

```
{  
  "propertyId":{"  
    "DeviceType":"Netvox_R718G_Uplink/types/DeviceTypeEnum",  
    "CmdId":"Netvox_R718G_Uplink/types/CmdIdEnum",  
    "Payload":"Netvox_R718G_Uplink/types/ConfigureCmdPayload"  
  }},  
  "propertyValue":{"  
    "DeviceType":"R718G",  
    "CmdId":"ConfigReportRsp",  
    "Payload":{"  
      "Status":"Success"  
    }  
  }  
}
```

h. API call (Read report configuration):

```
aws iotthingsgraph map-property \  
  --region us-east-1 \  
  --endpoint-url "https://iotthingsgraph.us-east-1.amazonaws.com" \  
  --namespace-snapshot file:///path/to/model/json/file \  
  --property-value "021E0000000000000000" \  
  --mapping-info '{"mappingId":"urn:tdm:us-east-1/000000000000/default:mapping:map_R718G_Uplink","mappingDirection": "FORWARD",  
"contextProvider":{"json": {"\"port\":7}}}'
```

i. Expected results

```
{  
  "propertyId": "{  
    "DeviceType": "Netvox_R718G_Uplink/types/DeviceTypeEnum",  
    "CmdId": "Netvox_R718G_Uplink/types/CmdIdEnum"  
  }",  
  "propertyValue": "{  
    "DeviceType": "R718G",  
    "CmdId": "ReadConfigReportReq"  
  }"  
}
```

j. Expected results (input data: 821E012C03840100000064)

```
{  
  "propertyId": "{  
    "DeviceType": "Netvox_R718G_Uplink/types/DeviceTypeEnum",  
    "CmdId": "Netvox_R718G_Uplink/types/CmdIdEnum",  
    "Payload": "Netvox_R718G_Uplink/types/ConfigureCmdPayload"  
  }",  
  "propertyValue": "{  
    "DeviceType": "R718G",  
    "CmdId": "ReadConfigReportRsp",  
    "Payload": {  
      "IlluminanceChange": 100,  
      "MaxTime": 900,  
      "MinTime": 300,  
      "BatteryChange": 0.1  
    }  
  }"  
}
```

k. API call (**Set Sunlight Sample Range**):

```
aws iotthingsgraph map-property \  
  --region us-east-1 \  
  --endpoint-url "https://iotthingsgraph.us-east-1.amazonaws.com" \  
  --namespace-snapshot file:///path/to/model/json/file \  
  --property-value "031E0200000000000000" \  
  --mapping-info '{"mappingId":"urn:tdm:us-east-1/000000000000/default:mapping:map_R718G_Uplink","mappingDirection": "FORWARD",  
"contextProvider":{"json": "{\\"port\\":7}"}}'
```

l. Expected results

```
{  
  "propertyId":{"  
    "DeviceType":"Netvox_R718G_Uplink/types/DeviceTypeEnum",  
    "CmdId":"Netvox_R718G_Uplink/types/CmdIdEnum",  
    "Payload":"Netvox_R718G_Uplink/types/ConfigureCmdPayload"  
  }},  
  "propertyValue":{"  
    "DeviceType":"R718G",  
    "CmdId":"SetSunlightSampleRangeReq",  
    "Payload":{"  
      "RangeSetting":"3 ~ 22000"  
    }  
  }  
}
```

m. Expected results (input data: 831E00000000000000000000)

```
{  
  "propertyId":{"  
    "DeviceType":"Netvox_R718G_Uplink/types/DeviceTypeEnum",  
    "CmdId":"Netvox_R718G_Uplink/types/CmdIdEnum",  
    "Payload":"Netvox_R718G_Uplink/types/ConfigureCmdPayload"  
  }},  
  "propertyValue":{"  
    "DeviceType":"R718G",  
    "CmdId":"SetSunlightSampleRangeRsp",  
    "Payload":{"  
      "Status":"Success"  
    }  
  }  
}
```

n. API call (**Get Sunlight Sample Range**):

```
aws iotthingsgraph map-property \  
  --region us-east-1 \  
  --endpoint-url "https://iotthingsgraph.us-east-1.amazonaws.com" \  
  --namespace-snapshot file:///path/to/model/json/file \  
  --property-value "041E0000000000000000" \  
  --mapping-info '{"mappingId":"urn:tdm:us-east-1/000000000000/default:mapping:map_R718G_Uplink","mappingDirection": "FORWARD",  
"contextProvider":{"json": "{\\"port\\":7}"}}'
```

o. Expected results

```
{  
  "propertyId": "{  
    "DeviceType": "Netvox_R718G_Uplink/types/DeviceTypeEnum",  
    "CmdId": "Netvox_R718G_Uplink/types/CmdIdEnum"  
  }",  
  "propertyValue": "{  
    "DeviceType": "R718G",  
    "CmdId": "GetSunlightSampleRangeReq"  
  }"  
}
```

p. Expected results (input data: 841E02000000000000000000)

```
{  
  "propertyId": "{  
    "DeviceType": "Netvox_R718G_Uplink/types/DeviceTypeEnum",  
    "CmdId": "Netvox_R718G_Uplink/types/CmdIdEnum",  
    "Payload": "Netvox_R718G_Uplink/types/ConfigureCmdPayload"  
  }",  
  "propertyValue": "{  
    "DeviceType": "R718G",  
    "CmdId": "GetSunlightSampleRangeRsp",  
    "Payload": {  
      "RangeSetting": "3 ~ 220000"  
    }  
  }"  
}
```

[Support](#)

For questions on this model, please contact: support@netvox.com.tw