

Getting started guide

Functionality

The detecting range of R718CT is -40 °C~ +125°C.

R718CT is more stable when detecting the temperature range of -40°C~0°C.

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_R718CT_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		
R718CT	0x92	0x01	Battery (1Byte, unit:0.1V)	Temperature (2 Bytes, signed unit:0.1°C)	Reserved (5 Bytes, fixed 0x00)

- d. Input binary data: 0192012400FD0000000000
 (DeviceType: 0x92, ReportType: 0x01, Battery: 24_{hex} = 36_{dec}, Temperature1: 00FD_{hex} = 253_{dec},
 Temperature2: 0109_{hex} = 265_{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 "0192012400FD0000000000" \
  --mapping-info '{"mappingId":"urn:tdm:us-east-1/000000000000/default:mapping:map_R718CT_Uplink","mappingDirection": "FORWARD",
  "contextProvider":{"json": "{\"port\":6}"}}'
```

- f. Expected results (V1)

```
{
  "propertyId": "Netvox_R718CT_Uplink/types/UplinkPort6Payload",
  "propertyValue": "{
    \"DeviceType\": \"R718CT\",
    \"Temperature\": 25.3,
    \"Version\": 1,
    \"Battery\": 3.6,
    \"ReportType\": 1
  }"
```

- 2. Test name: Set and read report configuration
 - a. Mapping used: map_R718CT_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	0x92	MinTime (2bytes Unit:s)	MaxTime (2bytes Unit:s)	Battery Change (1byte Unit:0.1v)	Temperature Change (2Bytes, Unit:0.1°C)	Reserved (2Bytes, Fixed 0x00)
Config ReportRsp	0x81		Status (0x00:success)	Reserved (8Bytes, Fixed 0x00)			
ReadConfig ReportReq	0x02		Reserved (9Bytes, Fixed 0x00)				
ReadConfig ReportRsp	0x82		MinTime (2bytes Unit:s)	MaxTime (2bytes Unit:s)	Battery Change (1byte Unit:0.1v)	Temperature Change (2Bytes, Unit:0.1°C)	Reserved (2Bytes, Fixed 0x00)

- d. Input binary data:

- i. **Set Report Configuration**

MinTime = 5min(300s), MaxTime = 15min(900s), BatteryChange = 0.1v, TemperatureChange = 1°C

Downlink: 0192012C038401000A0000 012C_{hex} = 300_{dec}, 0384_{hex} = 900_{dec},

0.1v(Unit:0.1v) => 0.1 ÷ 0.1 = 1, 01_{hex} = 1_{dec}

1°C(Unit:0.1°C) => 1 ÷ 0.1 = 10, 000A_{hex} = 10_{dec}

Response: 819200000000000000000000 (Configuration success)

819201000000000000000000 (Configuration failure)

- ii. **Read Report Configuration**

Downlink: 029200000000000000000000

Response: 8292012C038401000A0000 (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 "0192012C038401000A0000" \  
  --mapping-info '{"mappingId":"urn:tdm:us-east-  
1/000000000000/default:mapping:map_R718CT_Uplink","mappingDirection": "FORWARD",  
"contextProvider":{"json":{"port":7}}}'
```

f. Expected results

```
{  
  "propertyId":{"  
    "DeviceType":"Netvox_R718CT_Uplink/types/DeviceTypeEnum",  
    "CmdId":"Netvox_R718CT_Uplink/types/CmdIdEnum",  
    "Payload":"Netvox_R718CT_Uplink/types/ConfigureCmdPayload"  
  }},  
  "propertyValue":{"  
    "DeviceType":"R718CT",  
    "CmdId":"ConfigReportReq",  
    "Payload":{"  
      "TemperatureChange":1,  
      "MaxTime":900,  
      "MinTime":300,  
      "BatteryChange":0.1  
    }  
  }  
}"  
}
```

g. Expected results (input data: 819200000000000000000000)

```
{  
  "propertyId":{"  
    "DeviceType":"Netvox_R718CT_Uplink/types/DeviceTypeEnum",  
    "CmdId":"Netvox_R718CT_Uplink/types/CmdIdEnum",  
    "Payload":"Netvox_R718CT_Uplink/types/ConfigureCmdPayload"  
  }},  
  "propertyValue":{"  
    "DeviceType":"R718CT",  
    "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 "02920000000000000000" \  
  --mapping-info '{"mappingId":"urn:tdm:us-east-1/000000000000/default:mapping:map_R718CT_Uplink","mappingDirection": "FORWARD",  
"contextProvider":{"json": {"\"port\":7}}}'
```

i. Expected results

```
{  
  "propertyId":{"  
    "DeviceType":"Netvox_R718CT_Uplink/types/DeviceTypeEnum",  
    "CmdId":"Netvox_R718CT_Uplink/types/CmdIdEnum"  
  }},  
  "propertyValue": "{  
    "DeviceType":"R718CT",  
    "CmdId":"ReadConfigReportReq"  
  }"  
}
```

j. Expected results (input data: 8292012C038401000A0000)

```
{  
  "propertyId":{"  
    "DeviceType":"Netvox_R718CT_Uplink/types/DeviceTypeEnum",  
    "CmdId":"Netvox_R718CT_Uplink/types/CmdIdEnum",  
    "Payload":"Netvox_R718CT_Uplink/types/ConfigureCmdPayload"  
  }},  
  "propertyValue":{"  
    "DeviceType":"R718CT",  
    "CmdId":"ReadConfigReportRsp",  
    "Payload":{"  
      "TemperatureChange":1,  
      "MaxTime":900,  
      "MinTime":300,  
      "BatteryChange":0.1  
    }  
  }  
}
```

3. Test name: Sensor Calibrate

- a. Mapping used: map_R718CT_Uplink
- b. Use case tested: Sensor Calibrate Configuration
- c. Payload format: **port = 14**

Description	CmdID	Sensor Type	PayLoad (Fix = 9 bytes)				
SetGlobal CalibrateReq	0x01	0x01	Channel (1Byte) Channel1 : 0, Channel2 : 1, etc.	Multiplier (2bytes, Unsigned)	Divisor (2bytes, Unsigned)	DeltValue (2bytes, Signed)	Reserved (2Bytes, Fixed 0x00)
SetGlobal CalibrateRsp	0x81		Channel (1Byte) Channel1 : 0, Channel2 : 1, etc.	Status (1Byte, 0x00_success)		Reserved (7 Bytes, Fixed 0x00)	
GetGlobal CalibrateReq	0x02		Channel (1Byte) Channel1 : 0, Channel2 : 1, etc.	Reserved (8Bytes,Fixed 0x00)			
GetGlobal CalibrateRsp	0x82		Channel (1Byte) Channel1 : 0, Channel2 : 1, etc.	Multiplier (2bytes, Unsigned)	Divisor (2bytes, Unsigned)	DeltValue (2bytes, Signed)	Reserved (2Bytes, Fixed 0x00)
ClearGlobal CalibrateReq	0x03	Reserved (10 Bytes,Fixed 0x00)					
ClearGlobal CalibrateRsp	0x83	Status (1Byte, 0x00_success)	Reserved (9 Bytes,Fixed 0x00)				

d. Input binary data:

i. **Set Global Calibrate**

If the temperature the R718CT detects is 16° and the actual temperature is 26°, it means the calibration we want to make is +10°

Channel 1= 0x00, Multiplier = 0x0001, Divisor = 0x0001, DeltValue = 0x0064

Downlink: 0101000001000100640000

Response: 8101000000000000000000 (Configuration success)

8101010000000000000000 (Configuration failure)

ii. **Get Global Calibrate**

Downlink: 0201000000000000000000

Response: 8201000001000100640000 (Current configuration)

iii. **Clear Global Calibrate**

Downlink: 030000000000000000000000
Response: 830000000000000000000000 (Configuration success)
830100000000000000000000 (Configuration failure)

e. **API call(Set Global Calibrate):**

```
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 "0101000001000100640000" \  
  --mapping-info '{"mappingId":"urn:tdm:us-east-1/000000000000/default:mapping:map_R718CT_Uplink","mappingDirection": "FORWARD",  
"contextProvider":{"json": {"port":14}}}'
```

f. **Expected results**

```
{  
  "propertyId":{"  
    "SensorType":"Netvox_R718CT_Uplink/types/SensorTypeEnum",  
    "CmdId":"Netvox_R718CT_Uplink/types/GlobalCalibrateCmdEnum",  
    "Channel":"integer",  
    "Multiplier":"integer",  
    "DeltValue":"integer",  
    "Divisor":"integer"  
  }},  
  "propertyValue":{"  
    "SensorType":"TemperatureSensor",  
    "CmdId":"SetGlobalCalibrateReq",  
    "Channel":1,  
    "Multiplier":1,  
    "DeltValue":100,  
    "Divisor":1  
  }  
}
```

g. **Expected results (input data: 810100000000000000000000)**

```
{  
  "propertyId":{"  
    "Status":"Netvox_R718CT_Uplink/types/ConfigReportRspStatus",  
    "SensorType":"Netvox_R718CT_Uplink/types/SensorTypeEnum",  
    "CmdId":"Netvox_R718CT_Uplink/types/GlobalCalibrateCmdEnum",  
    "Channel":"integer"  
  }},  
  "propertyValue":{"  
    "Status":"Success",  
    "SensorType":"TemperatureSensor",  
    "CmdId":"SetGlobalCalibrateRsp",  
    "Channel":1  
  }  
}
```

h. API call(Get Global Calibrate):

```
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 "02010000000000000000" \  
  --mapping-info '{"mappingId":"urn:tdm:us-east-1/000000000000/default:mapping:map_R718CT_Uplink","mappingDirection": "FORWARD",  
"contextProvider":{"json": {"port":14}}}'
```

i. Expected results

```
{  
  "propertyId": "{  
    "SensorType": "Netvox_R718CT_Uplink/types/SensorTypeEnum",  
    "CmdId": "Netvox_R718CT_Uplink/types/GlobalCalibrateCmdEnum",  
    "Channel": "integer"  
  }",  
  "propertyValue": "{  
    "SensorType": "TemperatureSensor",  
    "CmdId": "GetGlobalCalibrateReq",  
    "Channel": 1  
  }"  
}
```

j. Expected results (input data: 8201000001000100640000)

```
{  
  "propertyId": "{  
    "SensorType": "Netvox_R718CT_Uplink/types/SensorTypeEnum",  
    "CmdId": "Netvox_R718CT_Uplink/types/GlobalCalibrateCmdEnum",  
    "Channel": "integer",  
    "Multiplier": "integer",  
    "DeltValue": "integer",  
    "Divisor": "integer"  
  }",  
  "propertyValue": "{  
    "SensorType": "TemperatureSensor",  
    "CmdId": "GetGlobalCalibrateRsp",  
    "Channel": 1,  
    "Multiplier": 1,  
    "DeltValue": 100,  
    "Divisor": 1  
  }"  
}
```

k. API call(Clear Global Calibrate):

```
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 "03000000000000000000" \  
  --mapping-info '{"mappingId":"urn:tdm:us-east-1/000000000000/default:mapping:map_R718CT_Uplink","mappingDirection": "FORWARD",  
"contextProvider":{"json": {"\"port\":14}}}'
```

l. Expected results

```
{  
  "propertyId":{"  
    "CmdId":"Netvox_R718CT_Uplink/types/GlobalCalibrateCmdEnum"  
  }},  
  "propertyValue":{"  
    "CmdId":"ClearGlobalCalibrateReq"  
  }"  
}
```

m. Expected results (input data: 8300000000000000000000)

```
{  
  "propertyId":{"  
    "Status":"Netvox_R718CT_Uplink/types/ConfigReportRspStatus",  
    "CmdId":"Netvox_R718CT_Uplink/types/GlobalCalibrateCmdEnum"  
  }},  
  "propertyValue":{"  
    "Status":"Success",  
    "CmdId":"ClearGlobalCalibrateRsp"  
  }"  
}
```

Support

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