



Device Services Contribution Proposal

Jiangxing Intelligence Inc.

2020.08.24

Danyang Song (Arthur)
Yaohong Li (Michael)

📍 Shenzhen Beijing Nanjing

EdgeX Foundry Project
https://www.edgexfoundry.org/

Repositories 102 Packages People 38 Projects 10

device

11 results for source repositories matching device

- device-sdk-go
- device-modbus-go
- device-mqtt-go**
- device-random

MQTT

Examples and Tutorials

- Provision a Device - Modbus Example
- Modbus - Data Type Conversion Using the Virtual Device Service
- Random Integer Device Service Example
- Sending and Consuming Binary Data From EdgeX Device Services
- App Service Examples
- Command Devices with Kuiper Rules Engine
- Adding a Device
 - Modbus
 - SNMP
 - MQTT
 - Walkthrough

1.2-Geneva

Add Device Service to docker-compose File (docker-compose.yml)

Download the Geneva release docker-compose file from <https://github.com/edgexfoundry/developer-scripts/blob/master/releases/geneva/compose-files/docker-compose-geneva-redis.yml>.

Because we deploy EdgeX using docker-compose, we must add device-mqtt to the docker-compose file. If you have prepared configuration files, you can mount them using volumes and change the endpoint for device-mqtt internal use.

This is illustrated in the following docker-compose file snippet:

```
device-mqtt:
  image: edgexfoundry/docker-device-mqtt-go:1.0.0
  ports:
    - "49982:49982"
  container_name: edgex-device-mqtt
  hostname: edgex-device-mqtt
  networks:
    - edgex-network
  volumes:
    - db-data:/data/db
    - log-data:/edgex/logs
    - consul-config:/consul/config
    - consul-data:/consul/data
    - ./mqtt:/custom-config
  depends_on:
    - data
    - command
  endpoint:
    - /device-mqtt
    - --registry=consul://edgex-core-consul:8500
    - --confdir=/custom-config
```

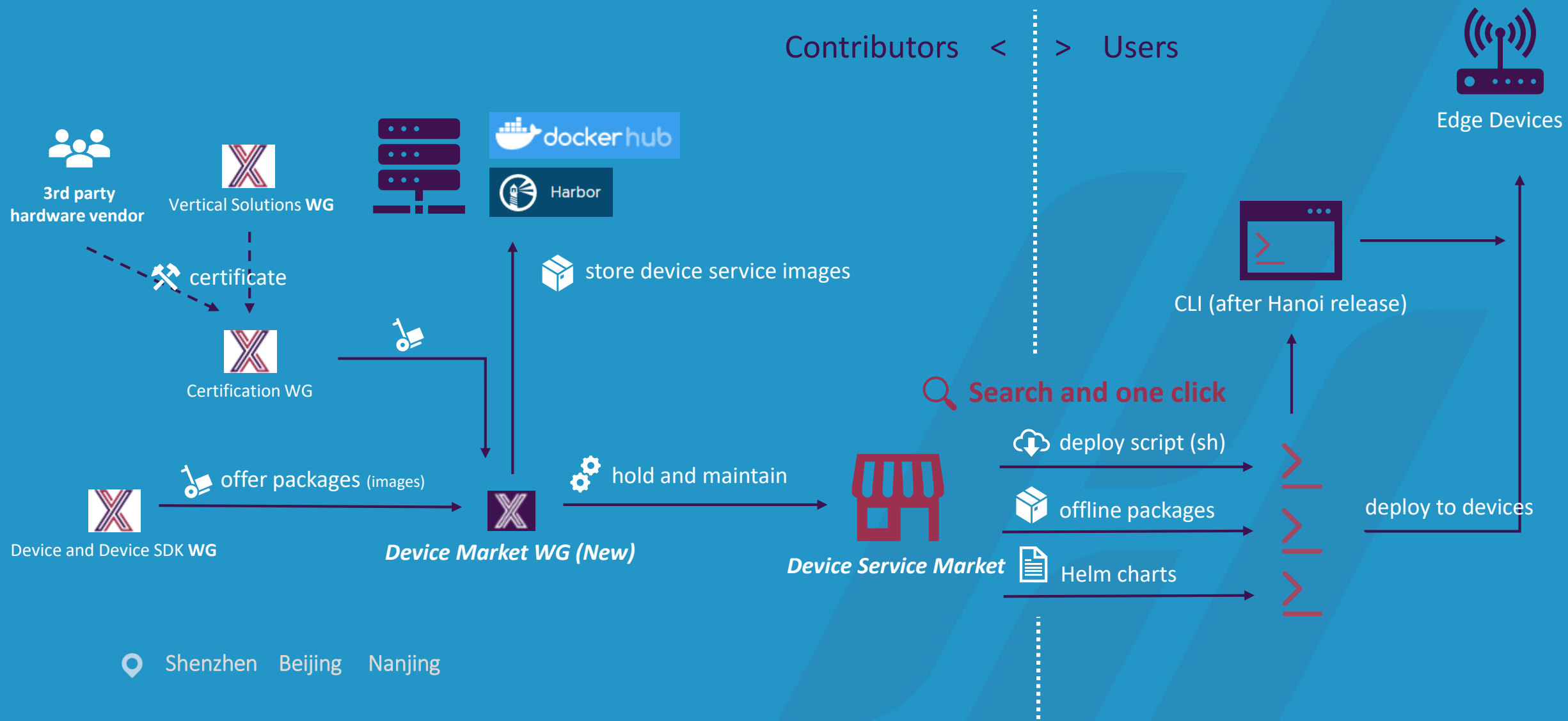


device-mqtt.yaml

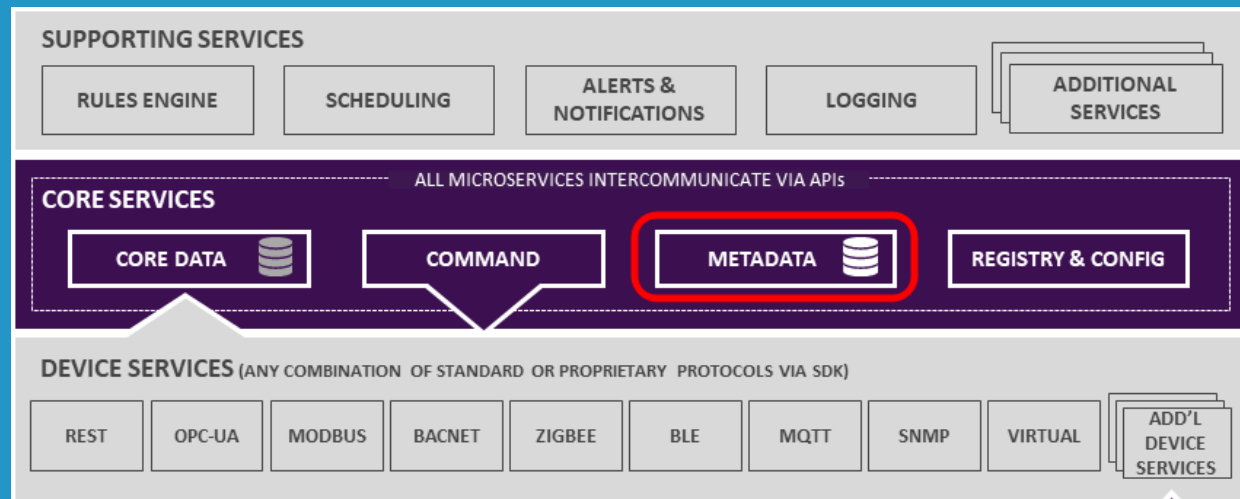


docker-compose up -d

Deployment w/ Online Market (Proposal)



Dataflow for users to deploy an EdgeX Foundry certificated device service

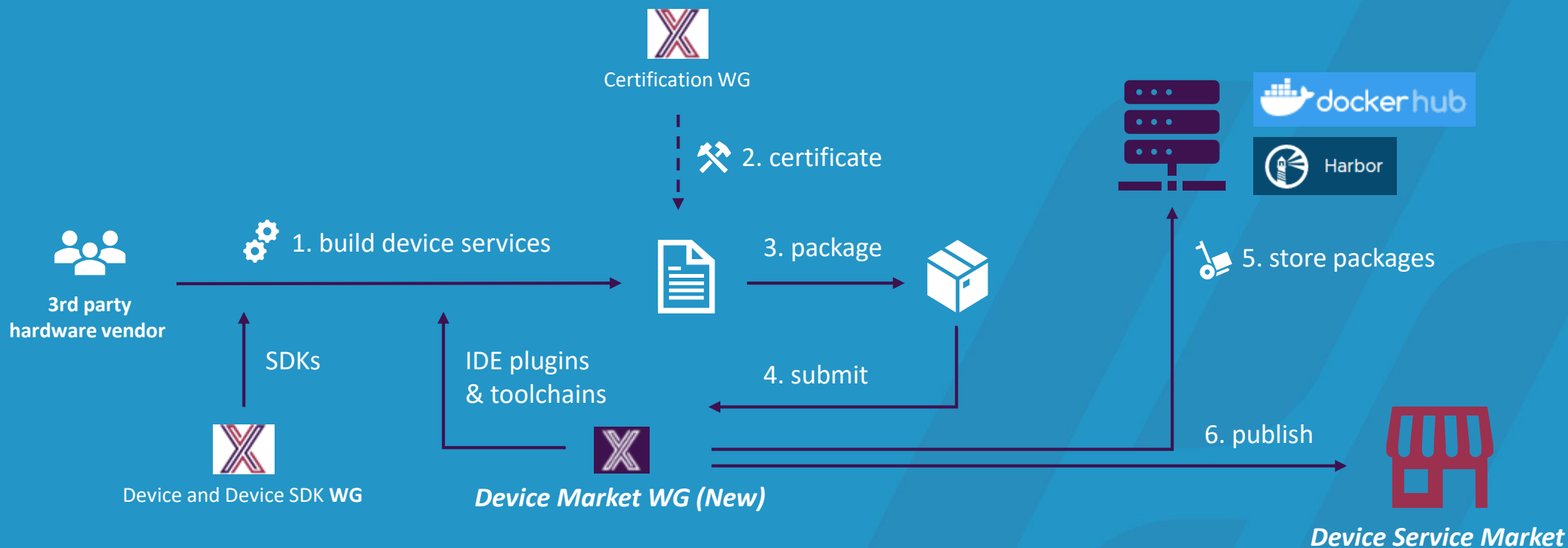


images pull

CLI (after Hanoi release)

DEPLOYED
DEVICE
SERVICES

Workflow for 3rd party hardware vendor / developer contribution (proposal)



IDE Plugins & Toolchains

1. device profile canonical form verification
2. scaffold project
3. documents and guideline integration
4. images builder
5. other languages binding (rust, etc.)
6. stress testing toolkit



Device and Device SDK WG



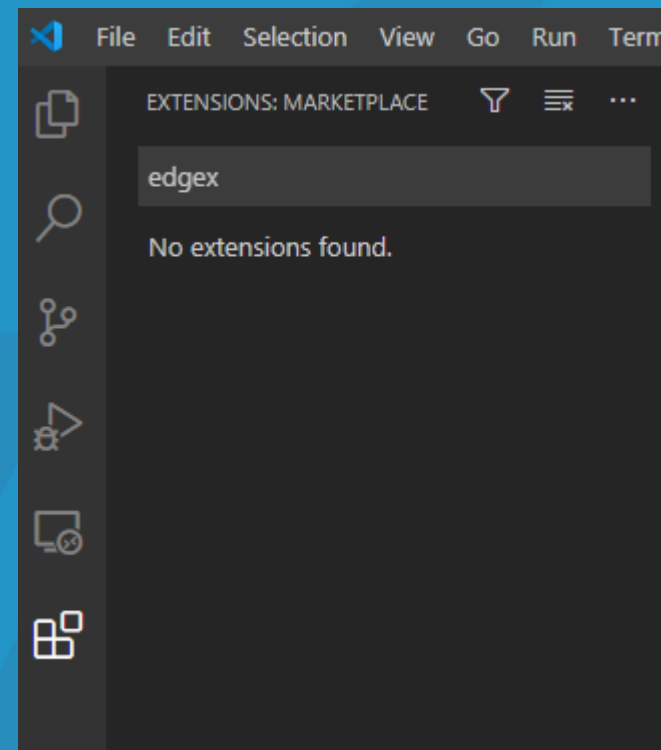
DevOps WG



QA/Test WG



Possibly integrate with CLI



9 device services contribution proposal

1. Buzzer alarm (sound alarm)
2. Infrared barrier module (obstacle detection)
3. PM2.5 detection (air quality)

4. 6-axis sensor (gyroscope, acceleration and angular tilt)
5. BDS/GPS positioning (positioning)
6. Ambient light sensor (illuminance)
7. Temperature sensor (temperature)

8. Bond button bistable relay (secondary circuit)
9. Universal serial port (generous purpose)

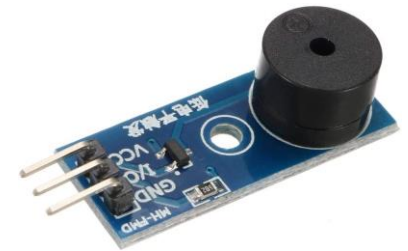
Passive Buzzer Alarm Module

Provides a buzzer sound and used for alarm or reminder

Vendor: Oiyagai [Amazon](#)

Voltage: DC 3.3V - 5V

Physical interface: GPIO*1, Input



Method	Core Command	Description	Response
PUT	open	keep beeping buzzer	2000K
PUT	close	stop beeping buzzer	2000K
PUT	beep	beeps buzzer for 1 second	2000K

Infrared Barrier Module

Determine whether there is an obstacle ahead. It can be widely used in obstacles avoidance, line counting, and black and white online tracking and so on

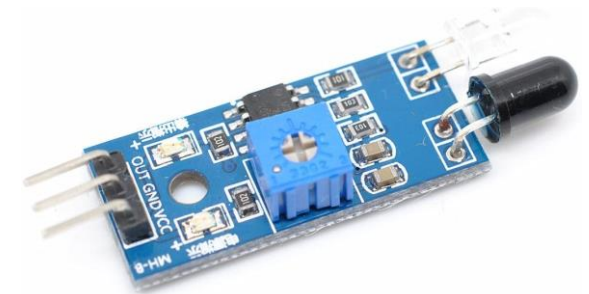
Vendor: Oiyagai [Amazon](#)

Voltage: DC 3V - 5V

Effective distance: 2cm - 30cm

Effective angle: 35°

Physical interface: GPIO*1, Output



Method	Core Command	Description	Response
GET	check	determine whether there is an obstacle ahead	"true" or "false"

ZPH02-PM2.5 Particles Module

Integrates infrared PM2.5 detection technology, using particle counting principle to detect PM2.5 in the environment

Vendor: [WINSSEN](#) [Amazon](#)

Voltage : DC 5V

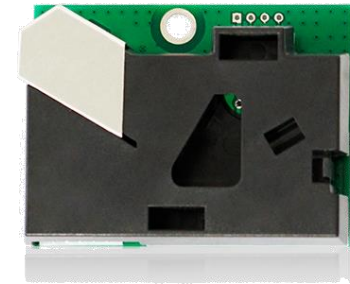
Particles diameter: $\geq 1\mu\text{m}$

Repeatability: 1 degree

Preheat time: about 1 minute

Operating temperature: 0 degrees Celsius ~ 50 degrees Celsius

Physical interface: TTL UART



Method	Core Command	Description	Response
GET	pm25	get value of PM2.5	"10" <string>

GY-25Z Serial Port MPU6050 Module

Get data of 6-axis sensor (gyroscope, acceleration and angular tilt)

Vendor: [Amazon](#)

Voltage: DC 3V – 5V

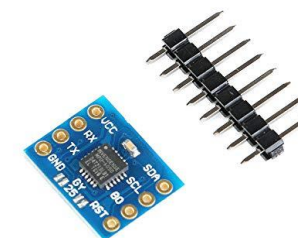
Measuring range: -180 degrees to 180 degrees

Resolution: 0.01 degrees

Measurement accuracy: 1 degree (inclination)

Repeatability: 1 degree

Physical interface: TTL UART



Method	Core Command	Description	Response
GET	light	get data of 6-axis sensor	{ "pitch":1.483, <float> "roll":-10.561, <float> "yaw":-8.887 <float> }

GP-02 GPS + BDS Compass ATGM336H-5N Module

Support single-system positioning of BDS/GPS/GLONASS satellite navigation system and the receiver module of any combination of multi-system joint positioning

Vendor: [icofchina](#) [Amazon](#)

Voltage: DC 2.7V - 3.6V

#channel: 32

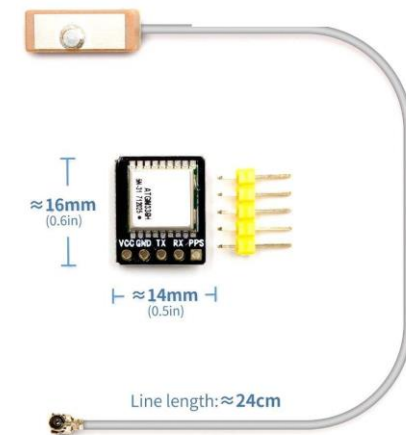
Cold start sensitivity: -148dBm

Physical interface: TTL

Dissipation Power: 25mA

Positioning accuracy: 2.5m (CEP50, open area)

Tracking sensitivity: -162dBm



Method	Core Command	Description	Response
GET	gps	get current location	<pre>{ "Altitude": 51.5, "Latitude": "22° 33' 6.100200", "Longitude": "113° 52' 55.000200", "NumSatellites": 7, "Time": "07:40:10.0000" }</pre>

GY-49 MAX44009 Ambient Light Sensor Module

Detect ambient light intensity

Vendor: [Amazon](#)

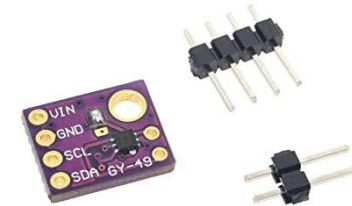
Voltage: DC 1.7V – 3.6V

Temperature Range: -40°C to +85°C

Effective angle: 35°

Dynamic range: 0.045 to 188,000 Lux

Physical interface: I2C



Method	Core Command	Description	Response
GET	light	get ambient light intensity	"383.040009 lux"

DS18B20 Temperature Sensor Module

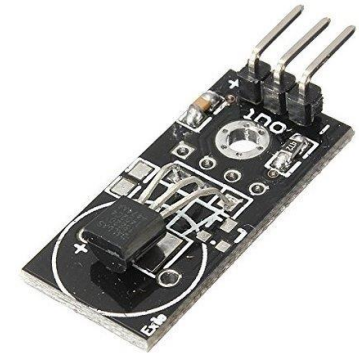
Measures ambient temperature

Vendor: [Amazon](#)

Voltage: DC 5V

Temperature measurement range: -55 degrees Celsius ~ +125 degrees Celsius

Physical interface: GPIO*1



Method	Core Command	Description	Response
GET	temperature	get temperature, unit 0.001 degrees Celsius	27000 <string>

Bond Button Bistable Relay Module



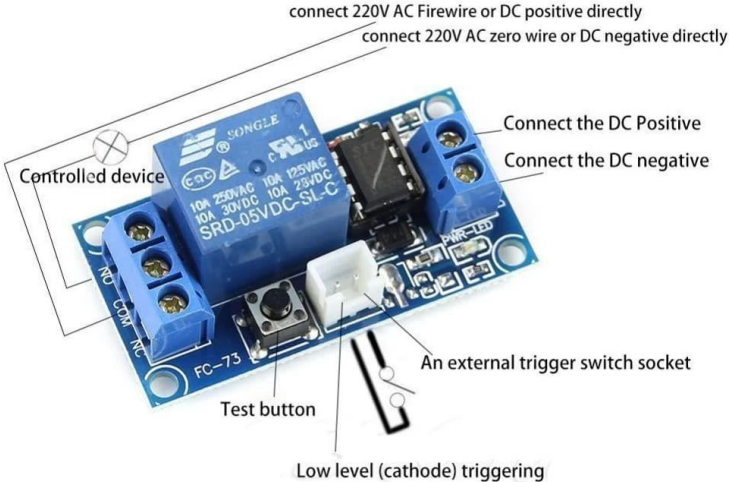
Controls the connection status of the secondary circuit(open circuit or close circuit)

Vendor: [Amazon](#)

Voltage: DC 5V

Relay secondary output end bears current: < 10A

Physical interface: GPIO*1



Method	Core Command	Description	Response
PUT	open	set the relay to close status	2000K
PUT	close	set the relay to open status	2000K

Universal serial port for Linux Module



Sends data, receives data and modifies the serial port configuration to any serial device for linux system, such as chip serial port, USB to TTL module, RS232 module and RS485 module

Vendor: [Amazon](#)

Physical interface: TTL, RS232, RS485, USB to TTL (UART)



Method	Core Command	Description	Response
GET	gethex	fetch serial device data, output in HEX string format	"{"rx utc": <int64>, "rxbuf hex":<string>}"
PUT	sendhex	send hex data to serial device	"{"sendhex":<string>}"
GET	getstring	fetch serial device data, output in ASCII string format	"{"rx utc": <int64>, "rxbuf hex":<string>}"
PUT	sendstring	send string data to serial device	"{"sendstring":<string>}"
GET	uartconfig	get current configuration	"{"baud":<string>, "path":<string>}"
PUT	uartconfig	set baud rate and configure serial device path on host	"{"baud":<string>, "path":<string>}"



EdgeX Foundry has been successfully applied by Jiangxing!

- 12+ industrial solutions
- 900+ production-level devices running insides

Almost all our devices used EdgeX and bring advantages

- 90% devices pre-installed with Jiangxing's released version of EdgeX
- More compatible products
- Flexible interfaces to swiftly extend

Willing to make more contributions and willing to maintain

- 10+ DevOps working on Jiangxing's released version
- Continuously stay connected with community
- Contribute more microservices, modules or device services
- Proposal on maintaining an industrial device services group

THANKS