江行智能

Device Services Contribution Proposal

Jiangxing Intelligence Inc.

2020.09.18

Danyang Song (Arthur)
Yaohong Li (Michael)

Shenzhen Beijing Nanjing

江行智能

Details of device service contributions

1. Two device service (GPIO & UART)
2. Device service examples

3. Current progress and questions
4. Next steps?

Two device service (GPIO & UART)

江行智能

device-gpio-go

Function:

This Device Service is a reference example of a Device Service developed with **Go Device Service SDK**

It is developed to control the system gpio through the following functions. For example, export or unexport the gpio, set gpio direction to output or input mode, set the gpio output status and get gpio status.

Physical interface: system gpio (/sys/class/gpio)

Driver protocol: IO

Usage:

This Device Service have to run with other EdgeX Core Services, such as Core Metadata, Core Data, and Core Command.

After starting the service, user can use "exportgpio" command to export a system gpio which comes from the path "/sys/class/gpio". Then configure the gpio by sending "gpiodirection" or "gpiovalue" command and check it's configuration by getting them.

PR: WIP: GPIO and UART device service contributions from J.I. team
https://github.com/edgexfoundry/edgex-examples/pull/10

Method	Core Command	Parameters	Description	Response
PUT	exportgpio	{export: <gpionum>}	export a gpio from /sys/class/gpio	200OK
PUT	unexportgpio	{unexport: <gpionum>}	unexport a gpio	200OK
PUT	gpiodirection	{direction: <direction>}	set direction of the exported gpio	200OK
GET	gpiodirection		get direction of the exported gpio	{direction: in, gpio: 65}
PUT	gpiovalue	{value: <value>}	set value for the exported gpio	200OK
GET	gpiovalue		get value for the exported gpio	{gpio:65, value: 1}

Two device service (GPIO & UART)

江行智能

device-uart-go

Function:

This Device Service is a reference example of a Device Service developed with **Go Device Service SDK**

It is developed for universal serial device, such as USB to TTL serial port, rs232 interface and rs485 interface device. It provides REST API interfaces to communicate with serial sensors or configure them

Physical interface: TTL, RS232, RS485, USB to TTL

Driver protocol: UART

Usage:

This Device Service have to run with other EdgeX Core Services, such as Core Metadata, Core Data, and Core Command.

After starting the service, the linux uart device will be opened and configured(defalut: /dev/ttyUSB0 with 115200 bps). User can choose another uart device by sending "uartconfig" command.

When uart device is opened, the data sent from another sensors will be stored and they can be read by "gethex" or "getstring" command. User can also send down link data to those sensors by "sendhex" or "sendstring" command.

PR: WIP: GPIO and UART device service contributions from J.I. team
https://github.com/edgexfoundry/edgex-examples/pull/10

Method	Core Command	Parameters	Description	Response
GET	gethex		get data, output in hex string	{rxbuf hex: 1A2B3C4D5E6F}
GET	getstring		get data, output in ASCII string	{rxbuf string: example-string}
PUT	sendhex	{sendhex: <txbuf>}	send data, input with hex string	200OK
PUT	sendstring	{sendstring: <txbuf>}	send data, input with ASCII string	200OK
GET	uartconfig		get configs	{baud: 9600, device path: /dev/ttyUSB5}
PUT	uartconfig	{path: <path>, baud: <baud>}	config device's path and baud	200OK

江行智能

Device service example

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)

Passive Buzzer Alarm Module

江行智能

Provides a buzzer sound and used for alarm or reminder
Vendor: Oiyagai [Amazon]

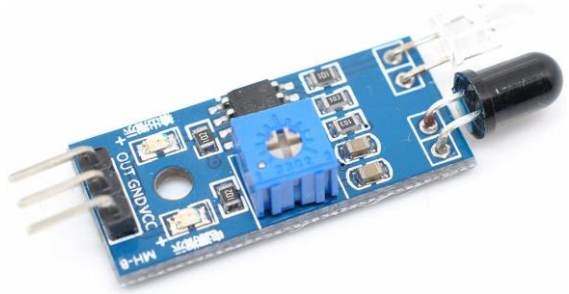
Voltage: DC 3.3V - 5V
Physical interface: GPIO*1, Input

Method	Core Command	Description	Response
PUT	open	keep beeping buzzer	200OK
PUT	close	stop beeping buzzer	200OK
PUT	beep	beeps buzzer for 1 second	200OK

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]

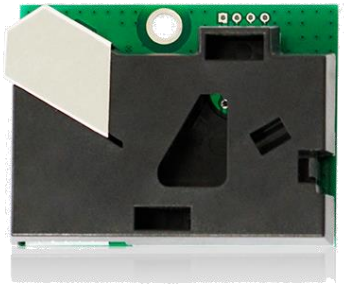
Voltage: DC 3V - 5V
Effective distance: 2cm - 30cm
Effective angle: 35°
Physical interface: GPIO*1, Output



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: WINSEN Amazon



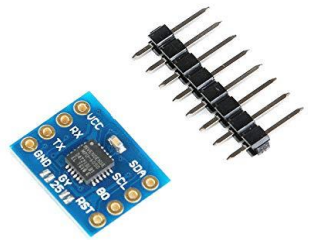
Voltage : DC 5V
Particles diameter: ≥1μm
Repeatability: 1 degree
Preheat time: about 1 minute
Operating temperature: 0 degrees Celsius ~ 50 degrees Celsius
Physical interface: TTL UART

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> }

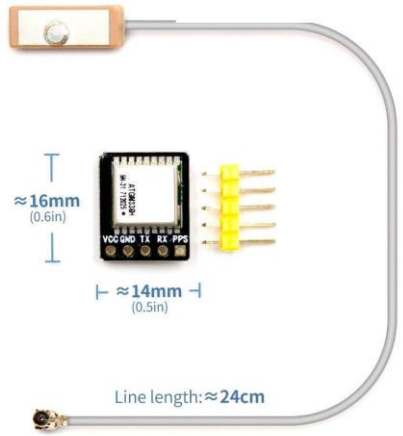
GP-02 GPS + BDS Compass ATGM336H-5N Module

江行智能

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	{ "Altitude": 51.5, <float> "Latitude": "22° 33' 6.100200", <string> "Longitude": "113° 52' 55.000200", <string> "NumSatellites": 7, <int> "Time": "07:40:10.0000" <string> }

GY-49 MAX44009 Ambient Light Sensor Module

Detect ambient light intensity
Vendor: Amazon

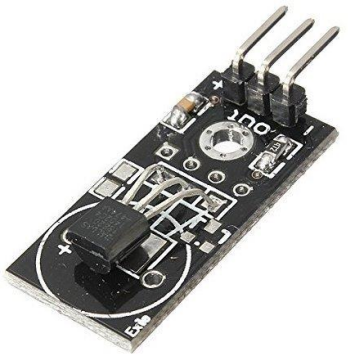
Voltage: DC 1.7V – 3.6V
Temperature Range: -40℃ to +85℃
Effective angle: 35°
Dynamic range: 0.045 to 188,000 Lux
Physical interface: I2C

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

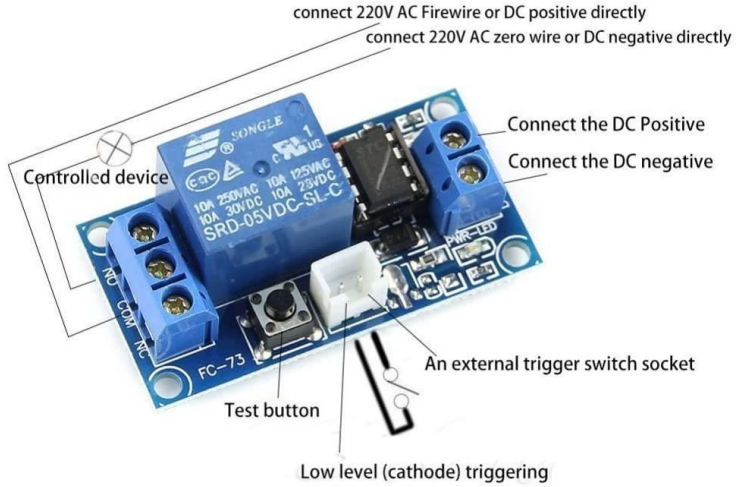


Method	Core Command	Description	Response
GET	temperature	get temperature, unit 0.001 degrees Celsius	27000 <string>

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	200OK
PUT	close	set the relay to open status	200OK

江行智能

Current progress

1. **PR: WIP: GPIO and UART device service contributions from J.I. team**
   https://github.com/edgexfoundry/edgex-examples/pull/10

Questions

1. **EdgeX Foundry docker-compose environment**
2. **UnitTests and CI?**
3. **DCO failure under PR /pull/10**
4. **Code review & License checking**
5. **Create repo under edgexfoundry for device-service**

Next Steps?


THANKS