

Cambrionix Hub API

Manual de usuario traducido

Cambrionix Hub API

1. Tabla de contenido

1. Tabla de contenido	1
2. Introducción	5
2.1. Instalación	6
2.2. requisitos previos	8
3. Inicio rápido	11
3.1. Ejemplo de Python	11
3.2. Ejemplo de mecanografiado	13
3.3. Ejemplo HTTP GET	14
3.4. Manejo de errores	15
4. API Call Structure	17
4.1. Objeto de solicitud JSON-RPC	17
4.2. Objeto de respuesta JSON-RPC	17
4.3. JSON-RPC Error Object	18
4.4. Call Structure	19
5. Métodos API	20
5.1. cbrx_apidetalles	22
5.2. cbrx_apiversion	26
5.3. certificado_cbrx	27
5.4. cbrx_certificate (conjunto)	28
5.5. cbrx_certificate (eliminar)	30
5.6. cbrx_certificate (get)	32
5.7. cbrx_config_set	35

5.8. cbrx_conexión_cli	37
5.9. cbrx_conexión_cerrar	39
5.10. cbrx_conexión_cerrar y bloquear	41
5.11. cbrx_conexión_get	43
5.12. cbrx_connection_getdictionary	45
5.13. cbrx_conexión_abierta	47
5.14. conjunto_conexión_cbrx	49
5.15. cbrx_connection_setdictionary	51
5.16. cbrx_conexión_desbloqueo	53
5.17. cbrx_device_get	55
5.18. cbrx_descubrir	57
5.19. cbrx_discover ("todos")	59
5.20. cbrx_discover_id_to_os_reference	63
5.21. cbrx_salir	65
5.22. encontrar_cbrx	66
5.23. cbrx_firmware	70
5.24. cbrx_firmware (añadir)	72
5.25. cbrx_firmware (lista)	74
5.26. cbrx_firmware (eliminar)	76
5.27. cbrx_firmware (estado)	78
5.28. cbrx_firmware (actualización)	81
5.29. cbrx_get_usb (tree)	83
5.30. cbrx_get_usb (descriptors)	88

5.31. cbrx_hub_get	93
5.32. cbrx_hub_set	95
5.33. cbrx_notificaciones	97
5.34. cbrx_pair_device	99
6. Notificaciones de la API	101
7. Métodos obsoletos	107
7.1. cbrx_apiversion (verdadero)	108
7.2. cbrx_get_usbtrees	112
8. Cadena de dispositivo	117
9. API Management	122
9.1. Stopping the API service	122
9.2. Starting the API Service	123
10. Información adicional	125
11. Inicio sesión	130
12. muelles	133
13. Dynamic Hubs	134
14. Diccionarios	135
14.1. Conjuntos de características	135
14.2. Obtener Diccionario	137
14.3. Establecer diccionario	237
14.4. Diccionarios obsoletos	294
15. Conexiones de enchufe	299
16. Control de los LED	300

17. Información de la batería	301
18. API Error codes	303

2. Introducción

The Cambrionix Hub API is designed to control Cambrionix products through a locally installed service called 'CambrionixApiService'. This service provides a programming interface to control Cambrionix units.

Se proporciona un contenedor de Python con una biblioteca JSON-RPC de dominio público que permitirá escribir scripts sin necesidad de estar demasiado familiarizado con JSON. Alternativamente, puede usar el lenguaje de programación de su elección para conectarse directamente al daemon a través de un socket TCP/IP estándar y enviar y recibir datos con formato JSON. Cuando la API se usa para comunicarse con un concentrador conectado a una red remota, esto se hace a través de un túnel SSH.

The Cambrionix Hub API supports multiple simultaneous client connections to itself and supports concurrent access to numerous hubs.

The Cambrionix Hub API is implemented in CambrionixApiService, which sits between the application and the Cambrionix units. Mapea las propiedades de las unidades Cambrionix en comandos API.

Puede descargar la última versión de este manual desde nuestro sitio web en el siguiente enlace. <https://www.cambrionix.com/cambrionix-api>

Biblioteca JSON-RPC

La API usa JSON-RPC sobre TCP. JSON-RPC es una conexión a JSON como formato de datos. JSON-RPC significa 'Llamada a procedimiento remoto de notación de objetos de JavaScript'. En resumen, JSON representa un formato ligero para el intercambio de datos. Es un formato para estructurar datos que es fácil de transferir.

Se puede utilizar cualquier lenguaje de programación que admita JSON-RPC; las bibliotecas están ampliamente disponibles para otros idiomas.

2.1. Instalación

Instalación de macOS®

Para macOS®, se proporciona un instalador que configurará CambrionixApiService para que se ejecute como un proceso daemon. El servicio está configurado para cargarse a pedido cuando está configurado, el puerto de escucha está conectado mediante launchd.

Due to how the Cambrionix Hub API runs, it requires User Account Control (UAC) interaction to be installed. This means that the installer cannot be run silently.

La instalación también puede realizar todos los pasos necesarios para configurar cualquier versión instalada de Python 2 y 3 y dirigirlo a varios scripts de ejemplo.

Instalación de Windows

Para Windows, se proporciona un instalador autoextraíble que configurará CambrionixApiService para que se ejecute como un servicio de Windows.

Due to how the Cambrionix Hub API runs, it requires User Account Control (UAC) interaction to be installed. This means that the installer cannot be run silently.

La instalación también puede realizar todos los pasos necesarios para instalar y configurar Python 2 y 3, y dirigirlo a varios scripts de ejemplo.

Instalación de Linux

El paquete Linux® viene como un paquete Debian que puede instalar a través de la GUI o desde la línea de comandos usando apt:

```
sudo dpkg -i /Downloads/cambrionix-api-setup-?????????.deb
```

You can also set the API to run in a persistent mode which will then run when you reboot your system this can be achieved by using the following command

```
sudo /usr/bin/CambrionixApiService --install --persistent
```

También se ha probado una versión armhf en oDroid y Raspberry Pi.

Actualmente no tenemos soporte para archivos rpm; Si necesita ayuda, puede obtener más información en la página a continuación.

<https://fedingo.com/how-to-convert-deb-to-rpm-files-in-linux/>

Controladores USB

The USB drivers that are required to access Cambrionix hubs are included with most OS's by default.

Estos controladores están incluidos en el instalador y se pueden instalar seleccionando el componente opcional durante la instalación. La opción no aparecerá si los controladores necesarios ya están presentes. Sin embargo, la instalación de estos controladores no se completa hasta que se conecta un cargador Cambrionix a la máquina host. If you install the API and the USB drivers before the first time you connect a Cambrionix charger, then the API will not start, and you will need to reboot the host machine after connecting a Cambrionix hub that will trigger the completion of the USB driver installation, to ensure that the API service is correctly started.

2.2. requisitos previos

Before using the Cambrionix Hub API, a few steps and checks need to be completed.

Acceso directo al hardware USB

For the API to retrieve USB information from connected devices, it must have direct access to the hardware. This means that running in a Virtual machines (VM) such as Parallels, VirtualBox and Microsoft Hyper-V are not supported as the virtualisation prevents the API from determining which USB device is connected to which physical port. Also, it is not unusual that such a virtual environment will not have access to serial devices necessary to communicate with the hub to query information.

Thunderbolt™ con Windows

You may need to update your Thunderbolt™ Bus Drivers and possibly the BIOS on Windows. Once the Thunderbolt™ device has been accepted to connect, you may need to turn it off and on again for Windows to connect physically

Cargador con capacidad de sincronización para información USB

For the API to return USB device information such as the VID, PID, Manufacturer, Description or Serial Number, there must be a USB connection from the host machine to the connected device. Esto solo está presente en productos con capacidad de sincronización. Charge-only products have a USB connection to the charger but not to connected devices. The API is functional with charge-only chargers but cannot return the USB device information.

Versión para productos de firmware universales

When used with this API, products using the universal firmware must have firmware version 1.52 or later installed. We recommend that the latest version is installed available from our website or through Cambrionix Connect; a table of all products and the firmware used is below.

Firmware	Número de parte	nombre del producto
Universal	PP15S	PowerPad15S
Universal	PP15C	PowerPad15C
Universal	PP8S	PowerPad8S
Universal	SS15	SuperSync15
Universal	TS3-16	ThunderSync3-16
ELEGANTE	TS3-C10	ThunderSync3-C10
Universal	Pala Sub-16	Pala Sub-16
Universal	U8S	U8S
Sincronización PD	PDSync-C4	PDSync-C4
Universal	ModIT-Max	ModIT-Max
Control de motor	Tablero de control de motores	ModIT-Max

controladores USB

The Cambrionix Hub API daemon (CambrionixApiService) must be able to communicate with the local hub. El concentrador aparecerá como un dispositivo USB. El dispositivo USB irá acompañado de un puerto de comunicaciones virtual (VCP). Las comunicaciones virtuales funcionan como un puerto de comunicaciones serie estándar, o puerto COM, como suele llamarse. El sistema operativo debe tener instalado el controlador VCP (Puerto COM virtual) adecuado.

Linux®	Mac OS	ventanas
El soporte predeterminado en el kernel es suficiente. No instale los controladores D2XX ya que esto entra en conflicto con los controladores VCP requeridos.	El soporte predeterminado en el sistema operativo es suficiente. No instale los controladores D2XX ya que esto entra en conflicto con los controladores VCP requeridos.	El soporte D2XX puede coexistir con el soporte VCP. Estos controladores se instalan automáticamente en las versiones más recientes de Windows 10.

Sistema operativo

Hemos probado el Cambrionix Hub API y puedo confirmar que los siguientes sistemas operativos funcionan con el Cambrionix Hub API . Es posible que existan otros sistemas operativos que funcionen pero que no hayan sido probados:

- Ventanas 10
- Windows 11
- macOS 11 (Big Sur)
- macOS 12 (Monterrey)
- macOS 13 (Ventura)
- macOS 14 (Sonoma)
- macOS 15 (Sequoia)
- Linux Ubuntu
- Linux Debian

Con Linux, solo realizamos pruebas utilizando el sistema operativo mencionado anteriormente. También se ha probado una versión ARM hard float (armhf) en oDroid y Raspberry Pi.

3. Inicio rápido

Algunos scripts de ejemplo en Node.JS, C#, VB.Net y Python se incluyen con los archivos instalados. Para Python, recomendamos usar el ejemplo asyncio más nuevo en lugar de los ejemplos síncronos más antiguos.

- Windows: %ProgramFiles%\Cambrionix\CambrionixAPI/examples
- Linux: /usr/local/share/cambrionix/apiservice/examples.
- macOS: /Librería/Cambrionix/ApiService/ejemplos.

Para cada tipo de código en la carpeta de ejemplos, deberá instalar los programas necesarios.

- Python 3.4 para los ejemplos de Python, así como Python 3.4, necesitará el módulo jsonrpc-websocket. Puede encontrar información de referencia sobre el uso de Python [aquí](#).
- Node.JS para el ejemplo de nodejs, también requerirá NPM o Yarn. Puede encontrar más información [aquí](#).
- Ejemplos de Visual Studio para C# o VB.Net

Para instalar el paquete Python async api, vaya a ejemplos/python/asyncio y ejecute

```
pip install .
```

3.1. Ejemplo de Python

El módulo jsonrpc-websocket convierte automáticamente el script de python en una solicitud JSON-RPC. El siguiente ejemplo es cómo el websocket convertirá un script simple.

Pitón.

```
cbrxapi.cbrx_connection_open("DJ000102")
```

Traducción JSON-RPC.

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_open",
  "params": [
```

```

    "DJ000102"
  ]
}

```

Replies are automatically converted back from JSON into a Python dictionary, or list or value as appropriate.

Aquí hay un ejemplo del uso de la API, el código está escrito en Python 3.6: El código de ejemplo no busca errores, por lo que el script de Python simplemente se detendrá en un error. El manejo robusto de errores de código debe incorporarse en su propio software.

```

# Import the cbrxapi library.
import sys
from cbrxapi import cbrxapi

print("Querying API Version...")
try:
    result = cbrxapi.cbrx_apiversion()
except Exception as e:
    print(f"Could not communicate with API : {e}")
    result = None

if result:
    print(f"API Version {result[0]}.{result[1]}")

# Call cbrx_discover with "local" to find any locally attached Cambrionix
units.
# This will return a list of local Cambrionix units.
print("Discovering local devices..")
result = cbrxapi.cbrx_discover("local")
if not result or len(result) == 0:
    print("No Cambrionix unit found.")
    sys.exit(0)

print(f"Discovered {len(result)} units")

for unit_id in result:
    serial_port = cbrxapi.cbrx_discover_id_to_os_reference(unit_id)

    try:
        # Open a connection to the Cambrionix unit, which will return a handle
        for
        # the connection.
        handle = cbrxapi.cbrx_connection_open(unit_id)
    except Exception as e:
        print(f"Could not open connection to {unit_id} : {e}")
        handle = None

    if handle:
        # Using the handle, get the "Hardware" and "nrOfPorts" properties

```

```

hardware = cbrxapi.cbrx_connection_get(handle, "Hardware")
n_ports = cbrxapi.cbrx_connection_get(handle, "nrOfPorts")

# Done using the Cambrionix unit, close the handle.
cbrxapi.cbrx_connection_close(handle)

# Finally, print out the information retrieved from the Cambrionix
unit.
print(f"* {hardware} on {serial_port} has {n_ports} ports")

```

3.2. Ejemplo de mecanografiado

Este es un ejemplo simple del uso de Typescript para usar la API para obtener información sobre los concentradores y dispositivos. Puede encontrar más información sobre TypeScript [aquí](#).

```

import React from 'react';
import WebSocket from 'react-websocket';

class MyApiInterface extends React.Component {
  lastId = 0;

  render() {
    return (
      <WebSocket ref={r => this.websocket = r} reconnect
        url="ws://localhost:43424" protocol="jsonrpc"
        onMessage={this.onDataReceived.bind(this)}
        onOpen={this.onApiConnection.bind(this)}
        onClose={this.onApiDisconnection.bind(this)} />
    );
  }

  requests = {};

  onDataReceived(json) {
    const data = JSON.parse(json);
    const id = data.id;
    if (id) {
      const request = this.requests[id];
      if (request && request.callback) {
        request.callback(data);
      }
      delete this.requests[id];
    }
    else
    {
      //Could get a notification here if you enable them on active connection
      //Notifications have no id and can arrive at any time
    }
  }
}

```

```

makeRequest(method, params, callback) {
  var packet = {
    jsonrpc: "2.0",
    id:      ++this.lastId,
    method:  method,
    params:  params,
  };

  this.requests[packet.id] = {packet: packet, callback: callback};

  this.websocket.sendMessage(JSON.stringify(packet));
}

onApiConnection() {
  console.log("Connected");
  this.makeRequest("cbrx_discover", ["local"], console.log);
}

onApiDisconnection() {
  console.log("Disconnected");
  this.requests = {}
}
}

```

3.3. Ejemplo HTTP GET

Las conexiones se pueden realizar directamente a un URI con el prefijo http, en cuyo caso el json se extrae de la propia dirección o del contenido del cuerpo de la solicitud GET. Puedes probar este ejemplo en tu navegador o desde la línea de comando, usando curl:

```

curl -get http://localhost:43424/?{"i-
d":0,"jsonrpc":"2.0","method":"cbrx_discover","params":["all"]}

```

Las conexiones de socket pueden ser datos binarios simples, solicitudes http GET o sockets web (como los de Node.js). Por ejemplo, pegar lo siguiente en la barra de direcciones de su navegador debería permitirle ver resultados rápidos:

```

http://localhost:43424/?{"jsonrpc":"2.0","id":1,"method":"cbrx_dis-
cover","params":["all"]}

```

Please be aware that on some Terminal/Command Prompt windows, you may find that you need to encode the URL to prevent error's from occurring.

Once encoded, the above URL should look like:

```
http://-
localhost:43424/%7B%22jsonrpc%22:%222.0%22,%22id%22:0,%22method%22:%22cbrx_api-
details%22%7D
```

3.4. Manejo de errores

Un error JSON-RPC devolverá un miembro de error que contiene los siguientes miembros:

- **código (obligatorio):** un número entero que indica un código de error JSON-RPC predefinido en el rango de -32768 a -32000 o un código de error CBRXAPI como se documenta en la sección Errores específicos de CBRXAPI.
- **mensaje (opcional):** una cadena de mensaje que explica el código de error
- **datos (opcional):** información adicional sobre el error, como mensajes de depuración o identificadores.

El JSON-RPC de Python utilizado provoca una excepción para una respuesta de error con la siguiente asignación:

- el código de miembro se devuelve en `e.error_code`
- el mensaje de miembro se devuelve en `e.error_message`
- los datos de los miembros se devuelven en `e.error_data`.

Puede capturar una respuesta de error con:

```
try:
    handle = cbrxapi.cbrx_connection_open(id)
except jsonrpc.RPCFault as e:
    gotException = True
    errorCode = e.error_code
    errorMessage = e.error_message
    errorData = e.error_data
```

Ejemplo de cómo crear un error y la respuesta que dará:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_open",
  "params": [
    "0"
  ]
}
```

Respuesta:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "error": {
    "code": -10001,
    "message": "ID not found"
  }
}
```

4. API Call Structure

Las descripciones de las llamadas API contienen solicitudes/respuestas JSON-RPC tal como las vería en el cable.

4.1. Objeto de solicitud JSON-RPC

JSON-RPC es un protocolo de llamada a procedimiento remoto (RPC) ligero y sin estado. Un JSON-RPC se representa mediante el envío de un objeto Solicitud. El objeto Solicitud tiene los siguientes miembros:

jsonrpc

Una cadena que especifica la versión del protocolo JSON-RPC. DEBE ser exactamente "2.0".

id

Un identificador establecido por el Cliente que DEBE contener un valor de Cadena, Número o NULO si se incluye. Si no se incluye, se supone que es una notificación.

método

Una cadena que contiene el nombre del método que se va a invocar.

parámetros

Un valor estructurado que contiene los valores de los parámetros que se utilizarán durante la invocación del método. Esto no es necesario para todos los métodos.

Agrupar todo esto dará la solicitud JSON-RPC completa:

```
{
  "jsonrpc": "version",
  "id": 0,
  "method": "method-name",
  "params": [
    "structured-params"
  ]
}
```

4.2. Objeto de respuesta JSON-RPC

Cuando se realiza una llamada de rpc, habrá una Respuesta, excepto en el caso de Notificaciones. La respuesta se expresa como un solo objeto JSON, con los siguientes

miembros:

jsonrpc

Una cadena que especifica la versión del protocolo JSON-RPC. DEBE ser exactamente "2.0".

id

Este miembro es el mismo que el valor del miembro id en el objeto de solicitud.

resultado

El valor de este miembro está determinado por el método en el objeto Solicitud.

error

Este miembro se devuelve solo en caso de error.

Agrupar todo esto dará la respuesta JSON-RPC completa:

```
{
  "jsonrpc": "version",
  "id": 0,
  "result": "method-result"
}
```

4.3. JSON-RPC Error Object

Cuando una llamada encuentra un error, el objeto de respuesta contendrá el miembro de error con un valor que es un objeto con los siguientes miembros:

código

Un número que indica el tipo de error que ocurrió. Este es un número entero.

mensaje

Una cadena que proporciona una breve descripción del error.

datos

Un valor primitivo o estructurado que contiene información adicional sobre el error.

Agrupar todo esto dará la respuesta JSON-RPC completa:

```
{ "jsonrpc": "version", "id": 0, "error": { "code": "error-code",  
  "message": "error-message" } }
```

4.4. Call Structure

Esta parte se eliminó de las secciones Sintaxis y Retorno a lo largo de este manual para simplificar la documentación.

Se deben pasar otros dos pares clave-valor para completar la solicitud JSON; Uno que indica la versión de JSON-RPC que se está utilizando, en este caso 2.0 y una identificación que identifica esta solicitud:

La identificación es obligatoria pero solo relevante si varias solicitudes pueden estar pendientes simultáneamente en la misma conexión. Ayuda a hacer coincidir las respuestas con las solicitudes (asíncronas). CambrionixApiService le dará la respuesta a una solicitud con la identificación coincidente.

```
{  
  "jsonrpc": "2.0",  
  "id": 0  
}
```

5. Métodos API

Hay 3 grupos de llamadas en la API:

- Versión - Obtener detalles sobre la API
- Descubrimiento: obtenga detalles sobre lo que está conectado a la API
- Conexión - Administrar conexiones y dispositivos conectados

Versión

Llamada API	Descripción
cbrx_apiversion	Obtener la versión de la API
cbrx_apidetalles	Obtenga una versión mejorada de los detalles de la API

Descubrimiento

Llamada API	Descripción
cbrx_descubrir	Descubre las unidades Cambrionix
cbrx_discover_id_to_os_reference	Asigne una ID de unidad de unidad a dispositivo tal como la usa el sistema operativo
encontrar_cbrx	Buscar dispositivos conectados a unidades Cambrionix locales
cbrx_get_usb (tree)	Devuelve todo el árbol USB que se ha descubierto.
cbrx_get_usb (descriptors)	Request entire dump of a USB device's descriptor information
cbrx_config_set	Establecer opciones de configuración

Conexión

Llamada API	Descripción
certificado_cbrx	Administrar certificados y claves privadas para la API
cbrx_conexión_abierta	Abrir una conexión a un concentrador

Llamada API	Descripción
cbrx_conexión_cerrar	Cerrar una conexión abierta a un concentrador
cbrx_connection_getdictionary	Obtener todas las claves en un concentrador especificado por el mango de conexión.
cbrx_conexión_get	Obtener una clave de un concentrador especificado por manija de conexión
cbrx_hub_get	Obtenga una clave de un concentrador especificado por el número de serie de los concentradores
cbrx_device_get	Obtenga una clave de un concentrador especificado por el número de serie del dispositivo USB
cbrx_connection_setdictionary	Enumere todas las teclas de comando y de escritura para el concentrador especificado por connectionHandle
conjunto_conexión_cbrx	Establecer una clave para el valor especificado en un concentrador especificado por connectionHandle
cbrx_hub_set	Establecer una clave para el valor especificado en un concentrador especificado por el número de serie de los concentradores
cbrx_notificaciones	Enviar notificaciones
cbrx_firmware	Agregar o eliminar archivos de firmware
cbrx_conexión_cerrar y bloquear	Cierre todas las conexiones a un concentrador y bloquéelo
cbrx_conexión_desbloqueo	Desbloquee un concentrador que estaba bloqueado previamente.
cbrx_conexión_cli	Realizar la operación de la interfaz de línea de comandos
cbrx_pair_device	Initiaite pairing of an iOS device
cbrx_salir	Reiniciar la API

5.1. cbrx_apidetalles

Devuelve una versión mejorada de los detalles de la API. Esta información también se puede obtener pasando un parámetro verdadero opcional a `cbrx_apiversion`.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_apidetalles"
}
```

Devoluciones:

```
{
  "resultado": {
    "versión": [ número de versión ],
    "semver": " variante-semver ", "commitid": número-commitid ,
    "sucursal": " nombre-sucursal ", "capacidad": [ capacidad-API ],
    "notificaciones": [ notificación-posible ], "instalar ":"
    ubicación de instalación ", "registro": " ubicación de registros
    ", "configuración": " ubicación de configuración ",
    "documentación": " ubicación de documentación ", "cpu": {
      "marca": " información de marca ", "arco": " arquitectura de CPU
      ", "características": [ características de CPU ], "núcleos":
      valor de núcleos
    },
    "os": " Información del sistema operativo "
  }
}
```

Producción	Descripción
<i>número de versión</i>	Número de versión de la API como un número entero (Mayor, Menor, Revisión, Compilación)
<i>variante semver</i>	El nombre completo de la versión de la API

Producción	Descripción
<i>variante-commitid</i>	El valor numérico del ID de confirmación
<i>nombre-sucursal</i>	La rama de la API instalada
<i>API-capacidad</i>	disponible con la versión de API
<i>notificación posible</i>	Matriz de cadenas para mostrar una posible notificación. ver Notificaciones de la API
<i>ubicación de instalación</i>	La ubicación de los archivos de instalación
<i>ubicación de registros</i>	La ubicación donde se almacenan los registros
<i>configuración-ubicación</i>	La ubicación de la configuración de la API
<i>documentación-ubicación</i>	La dirección web de la documentación de la API
<i>información de marca</i>	La marca de la CPU
<i>CPU-arquitectura</i>	La arquitectura de la CPU
<i>Características de la CPU</i>	Funciones disponibles en la CPU
<i>valor de núcleos</i>	Cuantos cores tiene la cpu
<i>os-información</i>	Sistema operativo que se ejecuta en la máquina local

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_apidetails"
}
```

Ejemplo de respuesta exitosa

```

{
  "jsonrpc": "2.0",
  "id": 0,
  "result": {
    "version": [
      3,
      7,
      0,
      34
    ],
    "semver": "3.7.0+34",
    "guid": {
      "id": "d0dc3cac-e165-4e38-88bb-39064431bdc9",
      "computerId": "35aea4bc-44d3-4e9e-9b3c-c33b965c5639"
    },
    "host": [
      {
        "ip": "10.167.111.81",
        "port": 0,
        "nameServer": "10.167.111.241",
        "domainName": "CBRX.LOCAL",
        "hostName": "CBRXPC-011",
        "adapterName": "Intel(R) Ethernet Controller (3) I225-V",
        "adapterType": "Ethernet"
      }
    ],
    "commitid": 4287981321,
    "branch": "release",
    "capability": [
      "protobuf",
      "crash-report",
      "notification"
    ],
    "notifications": [
      "usb-changed",
      "usb-device-attached",
      "usb-device-detached",
      "discover-changed",
      "dead-hub-changed",
      "firmware-progress",
      "rfid-received",
      "rfid-removed",
      "over-temperature",
      "over-voltage",
      "under-voltage",
      "certificate-changed"
    ],
    "install": "C:\\Program Files\\Cambrionix\\API",
    "logging": "C:\\ProgramData\\Cambrionix\\Log",
    "settings": "C:\\ProgramData\\Cambrionix",
    "documentation": "C:\\Program Files\\Cambrionix\\API\\Cambrionix Hub API Reference.html",
    "cpu": {
      "brand": "12th Gen Intel(R) Core(TM) i9-12900K",
      "arch": "x64",

```

```
    "features": [  
      "aes",  
      "avx",  
      "avx2",  
      "bmi1",  
      "bmi2",  
      "clflushopt",  
      "clfsh",  
      "clwb",  
      "cx16",  
      "cx8",  
      "erms",  
      "f16c",  
      "fma3",  
      "fpu",  
      "mmx",  
      "movbe",  
      "pclmulqdq",  
      "popcnt",  
      "rdrnd",  
      "rdseed",  
      "sha",  
      "smx",  
      "ss",  
      "sse",  
      "sse2",  
      "sse3",  
      "sse4_1",  
      "sse4_2",  
      "ssse3",  
      "tsc",  
      "vaes",  
      "vpclmulqdq"  
    ],  
    "cores": 24  
  },  
  "os": "Windows 10 Pro 21H2 Build 19044.1889 64-bit"  
}
```

5.2. cbrx_apiversion

Devuelve la versión de la API en ejecución.

Sintaxis: Ver API Call Structure

```
{ "method": "cbrx_apiversion" }
```

Hay otro método que se puede utilizar, consulte [cbrx_apidetalles](#) para más información.

Devoluciones:

```
{ "result": [Version-number] }
```

El *número de versión* consta de dos números separados por comas. El número más a la izquierda se llama mayor, el número más a la derecha se llama menor.

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "method": "cbrx_apiversion"  
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": [  
    3,  
    7  
  ]  
}
```

5.3. certificado_cbrx

Proporcione (o elimine) un certificado y una clave privada a la API para permitir conexiones SSL desde fuera de localhost (la máquina en la que se ejecuta la API). Sin este certificado, la API solo escuchará las conexiones en localhost:43424. Una vez que se proporcione un certificado válido y una clave privada, esto cambiará a 0.0.0.0:43424. Las conexiones externas (no desde localhost) solo se permitirán si son conexiones SSL (HTTPS o Secure WebSockets).

La API no realiza una copia del certificado ni de la clave privada ya que esto podría vulnerar la seguridad si se encuentran en carpetas de acceso limitado. El usuario con el que se ejecuta la API necesitará acceso a los archivos para poder utilizarlos. Todo esto se prueba cuando se emite el comando "establecer" y debería proporcionar suficiente información de error si no funciona.

Depende del usuario proporcionar un certificado que sea adecuado para su uso. Por ejemplo, si no está firmado por una autoridad de certificación, deberá tratar esto de la manera habitual, como firmar su certificado con su propia autoridad de certificación y agregarlo a su aplicación o navegador.

Con Google Chrome puedes usar esta [guía](#) .

Con Firefox puedes usar esta [guía](#) .

Con Safari puedes usar esta [guía](#) .

para otros navegadores hay guías que se pueden encontrar en línea.

Solo se admite una única configuración de certificado. Si se proporciona una contraseña, se ofusca por seguridad.

5.4. cbrx_certificate (conjunto)

Proporcione un certificado y una clave privada a la API.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_certificate",
  "parámetros": [
    "colocar", {
      "private-key": key-filename, "certificate": certificate-filename,
      "password": password
    }
  ]
}
```

Parámetro	Descripción
<i>nombre-archivo-clave</i>	The filename including the path of the private key
<i>nombre de archivo de certificado</i>	The filename including the path of the certificate
<i>contraseña</i>	contraseña opcional si lo requiere la clave privada

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Notas

- The files will need to be stored in a location that is accessible by the system and API.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_certificate",
  "params": [
    "set",
    {
      "private-key": "C:\\git\\capi\\cbrxjson\\certificate\\key.pem",
      "certificate": "C:\\git\\capi\\cbrxjson\\certificate\\cert.pem"
    }
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": true
}
```

5.5. cbrx_certificate (eliminar)

Elimina el certificado y la clave privada de la API:

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_certificate",
  "parametros": ["quitar"]
}
```

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_certificate",
  "params": [
    "remove"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": true  
}
```

5.6. cbrx_certificate (get)

Get the supplied certificate and private key information from the API.

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_certificate",  
  "params": "get"  
}
```

Devoluciones:

```
"resultado": {  
  "certificate": "Certificate",  
  "subject": {  
    "C": "Country",  
    "L": "Location",  
    "O": "Organisation",  
    "CN": "Common name"  
  },  
  "issuer": {  
    "C": "Country",  
    "O": "Organisation",  
    "CN": "Common name"  
  },  
  "serial_number": "Serial number",  
  "algorithm": "Algorithm",  
  "extensions": {  
    "subjectAltName": [Alternative names]  }  
}
```

```

    },
    "validity": {
    "not_after": Vaild until,
    "not_before": Valid from
    }

```

Variable	Descripción
<i>Certificate</i>	The public certificate in its entirety
<i>Country</i>	Country code
<i>Location</i>	Specific Location company is registered
<i>Organisation</i>	The organisations name
<i>Common name</i>	The name the organisation is referred to in the certificate
<i>Serial number</i>	Used to uniquely identify the certificate within a CA's systems
<i>Algorithm</i>	This contain a hashing algorithm and a digital signature algorithm. For example "sha256RSA" where sha256 is the hashing algorithm and RSA is the signature algorithm
<i>Alternative names</i>	All name associated with the certificate
<i>Valid until</i>	The time and date past which the certificate is no longer valid
<i>Valid from</i>	The earliest time and date on which the certificate is valid

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_certificate",
  "params": "get"
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": {
    "certificate": "-----BEGIN CERTIFICATE-----\r\nD.....CF7ig==\r\n-----END
CERTIFICATE-----\r\n",
    "subject": {
      "C": "GB",
      "L": "Cambridge",
      "O": "Cambrionix Limited",
      "CN": "*.api.cambrionix.com"
    },
    "issuer": {
      "C": "US",
      "O": "DigiCert Inc",
      "CN": "DigiCert TLS RSA SHA256 2020 CA1"
    },
    "serial_number": "096...9BFBE",
    "algorithm": "sha256WithRSAEncryption",
    "extensions": {
      "subjectAltName": [
        "*.api.cambrionix.com",
        "api.cambrionix.com"
      ]
    },
    "validity": {
      "not_after": 169...99,
      "not_before": 16...400
    }
  }
}
```

5.7. cbrx_config_set

Esta función permite establecer opciones de configuración persistentes. Multiple configuration keys can be set at once as the Example below shows.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_config_set",
  "parámetros": {
    " clave- configuración: valor-configuración
  }
}
```

Parámetro	Descripción
<i>clave de configuración</i>	La clave de configuración que desea cambiar, según la tabla a continuación.
<i>valor de configuración</i>	El valor al que desea cambiar la configuración.

Clave de configuración	Descripción
adb_path	El nombre de ruta completo al ejecutable ADB de las herramientas para desarrolladores de Android™.
actualización de batería habilitada	¿Se realizará la actualización de la batería?
batería-actualizada-concurrencia	Cuántas actualizaciones de batería simultáneas se ejecutarán simultáneamente.
batería-actualización-frecuencia-segundos	Cuántos segundos entre actualizaciones de batería.

Devoluciones:

```
{  
  "resultado": verdadero  
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "method": "cbrx_config_set",  
  "params": {  
    "battery-update-enabled": true,  
    "battery-update-concurrency": 2,  
    "battery-update-frequency-seconds": 60  
  }  
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": true  
}
```

5.8. cbrx_conexión_cli

Realice la operación de la interfaz de línea de comandos en el concentrador conectado y devuelva el resultado completo. Esto le permite ejecutar comandos directamente en la línea de comandos del concentrador sin detener el servicio API. This method is only for using the CLI commands to obtain information and not update settings, if you wish to change the internal hub settings please use the [Ajustes](#) command.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_connection_cli",
  "parámetros": [
    empuñadura de conexión
    comando CLI
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>comando CLI</i>	El comando CLI que desea enviar. Para todos los comandos de la CLI, consulte la documentación de la CLI. www.cambrionix.com/cambrionix-cli

Devoluciones:

```
{
  "resultado": [ cli-respuesta ] }
```

cli-response es una matriz de cadenas que contiene todas las líneas de salida devueltas por el comando. Para más información ver www.cambrionix.com/cli

Ejemplos

Ejemplo de solicitud JSON-RPC

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_cli",
  "params": [
    7654,
    "id"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": ["mfr:cambrionix,mode:main,hw:PP15S,hwid:0x13,fw:1.83,bl: 0.12,s-
n:000000,group:-,fc:un"]
}
```

5.9. cbrx_conexión_cerrar

Cierra una conexión a un concentrador abierto anteriormente, como se especifica en el identificador de conexión.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_connection_close",
  "parámetros": [ control de conexión ]
}
```

manija de conexión es el [Connection Handles](#) como un entero

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_close",
  "params": [
    7654
  ]
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": true  
}
```

5.10. cbrx_conexión_cerrar y bloquear

Cierre todas las conexiones a un concentrador y asegúrelo contra uso posterior hasta que lo suelte.

cbrx_conexión_desbloqueo. Otros procesos que usaban estas conexiones obtendrán errores si intentan acceder a este concentrador.

Sintaxis: ver API Call Structure

```
{  
  "method": "cbrx_connection_closeandlock", "params": [hub-serial]  
}
```

hub-serial is a string which is the serial number of the hub, each string is guaranteed to be unique.

Devoluciones:

```
{  
  "resultado": verdadero  
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

Ejemplo de solicitud JSON-RPC:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "method": "cbrx_connection_closeandlock",  
  "params": [  
    "DB0074F5"  
  ]  
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": true  
}
```

5.11. cbrx_conexión_get

Del concentrador especificado por el identificador de conexión, obtenga el valor de la clave.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    identificador de conexión , " clave de diccionario " ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>clave-diccionario</i>	como devuelto por una llamada a cbrx_connection_getdictionary ver Obtener Diccionario para más información

Devoluciones:

```
{
  "resultado": [ valor-diccionario ] }
```

valor-diccionario es el valor de la clave del diccionario, consulte [Obtener Diccionario](#) para más información.

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_get",
  "params": [
    569,
    "nrOfPorts"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 10
}
```

5.12. cbrx_connection_getdictionary

Obtenga todas las claves que pueden devolver información sobre el concentrador especificado.

Sintaxis: ver API Call Structure

```
{  
  "método": "cbrx_connection_getdictionary", "parámetros": [  
    control de conexión ] }
```

manija de conexión es el [Connection Handles](#) como un entero

Devoluciones:

```
{  
  "resultado": [ diccionario ] }
```

El *diccionario* es una matriz de cadenas que contienen los nombres de las claves y los valores del dispositivo. Por favor mira [Obtener Diccionario](#) sección.

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "method": "cbrx_connection_getdictionary",  
  "params": [  
    7654  
  ]  
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "identificación": 0,  
  "resultado": [ diccionario ]}
```

El *diccionario* es una matriz de cadenas que contienen los nombres de las claves y los valores del dispositivo. Por favor mira [Obtener Diccionario](#) sección.

5.13. cbrx_conexión_abierta

Abra una conexión con el concentrador especificado. Una apertura exitosa da como resultado un identificador de conexión que se puede usar para otras llamadas.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_connection_open",
  "parámetros": [
    hub-serial , location
  ]
}
```

Parámetros	Descripción
<i>hub-serial</i>	Este es el número de serie del hub devuelto de cbrx_descubrir
<i>ubicación</i>	consulte la tabla a continuación, si no se incluye, se establecerá de forma predeterminada en local

Parámetro de ubicación	Descripción
local	conectarse al centro local
muelles	conectarse a un concentrador en un muelles se debe especificar el id que se puede encontrar en cbrx_descubrir

Devoluciones:

```
{
  "resultado": [ identificador de conexión ] }
```

manija de conexión es el [Connection Handles](#) como un entero

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_open",
  "params": [
    "DB0074F5"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 579
}
```

5.14. conjunto_conexión_cbrx

En el concentrador especificado por el identificador de conexión, establezca el valor de la clave. Las llamadas a un muelle darán como resultado que la clave correspondiente se configure en ambos cargadores, excepto las claves específicas del puerto, que se dirigirán solo al cargador apropiado.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , " clave de diccionario ", valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>clave-diccionario</i>	como devuelto por una llamada a cbrx_connection_setdictionary ver Establecer diccionario para más información
<i>Valor</i>	El valor que desea establecer la clave también

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_set",
  "params": [
    5313,
    "TwelveVoltRail.OverVoltage",
    true
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": true
}
```

5.15. cbrx_connection_setdictionary

Enumere todas las claves de valor grabables y las claves de comando para el concentrador especificado por [Connection Handles](#) .

Sintaxis: ver API Call Structure

```
{  
  "método": "cbrx_connection_setdictionary", "parámetros": [  
    control de conexión ] }
```

manija de conexión es el [Connection Handles](#) como un entero

Devoluciones:

```
{  
  "resultado": [ diccionario ]  
}
```

El *diccionario* es una matriz de cadenas que contiene los nombres de las teclas de escritura y las teclas de comando del dispositivo. Por favor mira [Establecer diccionario](#) sección.

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "method": "cbrx_connection_setdictionary",  
  "params": [  
    7654  
  ]  
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "identificación": 0,  
  "resultado": [ diccionario ]  
}
```

Para el *diccionario* por favor vea [Establecer diccionario](#) sección

5.16. cbrx_conexión_desbloqueo

Desbloquee un concentrador que estaba bloqueado previamente.

Sintaxis: ver API Call Structure

```
{
  "method": "cbrx_connection_unlock", "params": [hub-serial]
}
```

hub-serial is a string which is the serial number of the hub, each string is guaranteed to be unique.

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

Ejemplo de solicitud JSON-RPC

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_unlock",
  "params": [
    "DB0074F5"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": true  
}
```

5.17. cbrx_device_get

Del concentrador especificado por el número de serie, obtenga el valor de la clave. Similar a [cbrx_conexión_get](#) Solo se aceptan los valores get que son relevantes para los puertos.

Tenga en cuenta que esto es más lento que otros métodos si necesita realizar varias operaciones en el mismo dispositivo.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_device_get",
  "parámetros": [
    usb-serial, dictionary-key
  ]
}
```

Parámetro	Descripción
<i>usb-serial</i>	USB serial number
<i>clave-diccionario</i>	como devuelto por una llamada a cbrx_connection_getdictionary ver Obtener Diccionario para más información

Devoluciones:

```
{
  "resultado": [ valor-diccionario ] }
```

valor-diccionario es el valor clave que se especifica ver [Obtener Diccionario](#) para más información.

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_device_get",
  "params": [
    "0000802000184C390CD2002E",
    "USBSpeed"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": "High"
}
```

5.18. cbrx_descubrir

Descubra las unidades Cambrionix, obtenga el número de serie del concentrador.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_discover",
  "parámetros": [ unidad ] }
```

Parámetro de unidad	Descripción
local	ID de unidad para concentrador adjunto
muelles	ID de unidad para múltiples concentradores conectados entre sí y adjuntos

Devoluciones:

```
{
  "result": [hub-serial] }
```

hub-serial is a string which is the serial number of the hub, each string is guaranteed to be unique.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_discover",
  "params": [
    "local"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": [  
    "000000897FD0505A"  
  ]  
}
```

5.19. cbrx_discover ("todos")

Descubra todas las unidades y devuelva información detallada sobre los concentradores y sus dispositivos conectados. Tenga en cuenta que solo se incluirán los dispositivos que aparezcan en un escaneo USB. A diferencia de los otros métodos de descubrimiento, en lugar de una matriz de números de serie, será un objeto de números de serie con contenido.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_discover",
  "parámetros": ["todos"] }
```

Devoluciones:

```
{
  "resultado": {
    "hub-serial": { "Status": "status", "Manufacturer":
    "manufacturer-name", "Firmware": "firmware-version",
    "Bootloader": "bootloader-version", "SerialNumber": "hub-
    serial", "Group": "group-order", "FormFactor": "firmware-type",
    "PanelID": hardware-id, "Hardware": "product-name",
    "HostSerialPort": "serial-port", "USBVersion": usb-version,
    "LocationID": location-ID, "nrOfPorts": port-quantity, "ExtPSU":
    external-PSU, "Uptime_sec": runtime, "Rebooted": reboot-flag,
    "SyncSupported": sync-possible, "FiveVolt": 5V-present,
    "TwelveVolt": 12V-present, "TemperatureMonitoring": temp-
    possible, "HardwareFlags": "hardware-flags", "Devices": {device-
    string}
      }
    }
  }
```

Producción	Descripción
<i>hub-serial</i>	Este es el número de serie del hub devuelto de cbrx_descubrir

Producción	Descripción
<i>estado</i>	si hay un puerto serie abierto, consulte Estado
<i>Nombre del Fabricante</i>	Nombre definido del fabricante, por defecto es 'Cambrionix'
<i>versión de firmware</i>	Número de versión del firmware. Formato 'N.nn'
<i>versión del gestor de arranque</i>	Número de versión del gestor de arranque. Formato 'N.nn'
<i>orden de grupo</i>	Se utiliza para ordenar concentradores, lo cual es útil cuando se actualizan productos conectados para que los productos posteriores se actualicen y reinicien primero.
<i>tipo de firmware</i>	Se utiliza para indicar qué firmware acepta el producto
<i>Identificación del hardware</i>	número de identificación de hardware del producto del panel frontal
<i>nombre del producto</i>	Nombre de hardware del producto
<i>Puerto serial</i>	El Puerto serial el producto está conectado.
<i>versión usb</i>	El número de versión USB de la conexión al concentrador. Formato 'N.nn'
<i>ubicación-ID</i>	El Location IDs como un entero
<i>cantidad portuaria</i>	Cuántos puertos tiene el producto
<i>fuentes de alimentación externa</i>	Si el producto tiene una fuente de alimentación externa
<i>tiempo de ejecución</i>	Cuánto tiempo ha estado encendido el producto (en ms). Sin límite

Producción	Descripción
<i>bandera de reinicio</i>	Si el indicador de reinicio es verdadero o falso
<i>sincronización posible</i>	Si el producto es capaz de sincronizar. verdadero o falso
<i>5V-presente</i>	Si el producto tiene suministro de 5V. verdadero o falso
<i>12V-presente</i>	Si el producto tiene suministro de 12V. verdadero o falso
<i>posible temporal</i>	Si el producto es lata de temperatura del monitor. verdadero o falso
<i>banderas de hardware</i>	Indicadores de hardware como se detalla en Obtener Diccionario
<i>dispositivo-cadena</i>	Información de los dispositivos conectados, consulte Cadena de dispositivo

Ejemplo

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_discover",
  "params": [
    "all"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": {
    "000000897FD0505A": {
      "Status": "active",
      "Manufacturer": "cambrionix",
      "Firmware": "1.88.0",
      "Bootloader": "0.21",
    }
  }
}
```

```
"SerialNumber": "000000897FD0505A",  
"Group": "-",  
"FormFactor": "un",  
"PanelID": 48,  
"Hardware": "SuperSync15",  
"HostSerialPort": "COM3",  
"USBVersion": 2.1,  
"LocationID": 574750720,  
"USB3CompanionLocationID": 2723151872,  
"HostPortLocationID": 574771200,  
"nrOfPorts": 15,  
"ExtPSU": true,  
"Uptime_sec": 167551,  
"Rebooted": true,  
"SyncSupported": true,  
"FiveVolt": true,  
"TwelveVolt": true,  
"TemperatureMonitoring": true,  
"HardwareFlags": "SLET",  
"Devices": {}  
}  
}  
}
```

5.20. cbrx_discover_id_to_os_reference

Asigne una ID de unidad para un concentrador descubierto a un nombre de dispositivo como lo usa el sistema operativo. Esto solo se puede usar para productos Cambrionix conectados localmente.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_discover_id_to_os_reference",
  "params": "[hub-serial]"
}
```

hub-serial is a string which is the serial number of the hub, each string is guaranteed to be unique.

Devoluciones:

```
{
  "resultado": [" nombre-dispositivo "]
}
```

device-name is what the OS uses for the connection that the hub which is identified by the *hub-serial*. Para obtener más información, consulte [Puerto serial](#)

Ejemplo:

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_discover_id_to_os_reference",
  "params": [
    "DB0074F5"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": "COM9"  
}
```

5.21. cbrx_salir

Reinicie la API.

Sintaxis: Ver API Call Structure

```
{  
  "método": "cbrx_exit"  
}
```

Devoluciones:

```
{  
  "resultado": verdadero  
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "method": "cbrx_exit"  
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": true  
}
```

5.22. encontrar_cbrx

Busque dispositivos conectados a unidades Cambrionix locales.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_find",
  "parámetros": [ ID ]
}
```

La *identificación* puede ser cualquiera de las siguientes formas:

Parámetro de ID	Descripción
VÍDEO	Busque cualquier dispositivo que coincida con la ID del proveedor. Mostrado como un entero
VÍDEO Y PID	Busque dispositivos que coincidan exactamente con los ID de proveedor y producto. Mostrado como un entero
NOMBRE	Busque cualquier cosa que coincida con la expresión regular proporcionada. The regex is run against a string made up of the manufacturer, product name, USB serial number and DeviePath (For an iPhone this is the UDID). "<nombre del fabricante>\x1D<nombre del producto>\x1D<número de serie>". La expresión regular se realiza como una búsqueda, en lugar de una coincidencia, por lo que no necesita hacer cosas como ".*iPhone.*" para hacer coincidir las subcadenas; "iPhone" es suficiente. Puedes ser tan estricto como quieras. Además, si se ha detectado la identidad del teléfono o el número de serie interno, estos también coincidirán.

Devoluciones:

```
{
  "resultado": {
    "usb-serial": {
      "HostDevice": "hub-serial", "HostPort": device-port,
```

```

"HostDescription": "product-name", "HostSerial": "serial-port",
"Device": ["device-string"] }

}

}

}

```

Producción	Descripción
<i>usb-serial</i>	El número de serie del dispositivo.
<i>hub-serial</i>	Este es el número de serie del hub devuelto de cbrx_descubrir
<i>dispositivo- puerto</i>	Número de puerto del concentrador al que también está conectado el dispositivo
<i>nombre del producto</i>	Nombre de hardware del producto
<i>Puerto serial</i>	El Puerto serial el producto está conectado.
<i>dispositivo- cadena</i>	Información de los dispositivos conectados, consulte Cadena de dispositivo

Los datos devueltos se ingresan en el número de serie de cualquier dispositivo que coincida con los criterios de búsqueda. El valor de cada nodo contiene detalles de la ubicación y los detalles exactos del dispositivo.

Se buscan los elementos especificados en todo el árbol USB y, si se encuentran en algún lugar debajo de un concentrador Cambrionix, se devolverán los detalles de la conexión. Esto sería especialmente útil para los dispositivos que están conectados a un dispositivo concentrador intermedio en lugar de estar conectados directamente al concentrador Cambrionix, como un teléfono con batería extendida y ranuras USB adicionales.

For any search results that do not have their own USB serial number, there will be an additional entry of NoSerial that is an array of such results, see information in below example.

Ejemplo

Ejemplo de solicitud JSON-RPC:

```

{

```

```

"jsonrpc": "2.0",
"id": 0,
"method": "cbrx_find",
"params": [
  "i(Phone|Pad)"
]
}

```

Ejemplo de respuesta exitosa

```

{
  "jsonrpc": "2.0",
  "id": 0,
  "result": {
    "974a9d1e6848316264a8a9d8b094b7d5e63a7ae5": {
      "HostDevice": "60003",
      "HostPort": 2,
      "HostDescription": "TS3-C10",
      "Device": {
        "VID": 1452,
        "PID": 4779,
        "Manufacturer": "Apple Inc.",
        "Description": "iPad",
        "SerialNumber": "974a9d1e6848316264a8a9d8b094b7d5e63a7ae5",
        "DeviceType": "Apple",
        "LocationID": 856686592,
        "DevicePath": "\\?\\usb#vid_1234&pid_5678#{a5dcbf10-653004fb951ed}",
        "USBVersion": 2,
        "USBPower": {
          "State": "D0",
          "Description": "On"
        },
      },
      "USBSpeed": {
        "Speed": "480Mbps",
        "Description": "High"
      },
      "Endpoints": {
        "Active": 6,
        "Maximum": 8,
        "Memory": 32768
      },
      "Battery": {
        "DataSource": "imobiledevice",
        "TrustLevel": "paired",
        "PairingSupported": true,
        "CurrentLevel": 100,
        "CurrentTime": 1663145986,
        "StartingLevel": 100,
        "StartingTime": 1663145986,
        "CapacityNew": 11560,
        "Capacity": 11441,
        "ChargingStatus": "full",
      }
    }
  }
}

```


5.23. cbrx_firmware

Los métodos de firmware pueden controlar todos los aspectos de la actualización del firmware en los concentradores Cambrionix. Hay varios subcomandos que le permiten agregar o eliminar archivos de firmware del almacenamiento local de la API, enumerar los archivos de firmware disponibles actualmente, actualizar el firmware de los archivos proporcionados y verificar el estado de las actualizaciones de firmware existentes.

The firmware type for can be one of “un” for Universal, “pd” for PDSync, “st” for the TS3-C10 or “mc” for motor control board. For information on your hubs requirements you can use the [Firmware Requirements](#) API method.

Firmware	Número de parte	nombre del producto
Universal	PP15S	PowerPad15S
Universal	PP15C	PowerPad15C
Universal	PP8S	PowerPad8S
Universal	SS15	SuperSync15
Universal	TS3-16	ThunderSync3-16
ELEGANTE	TS3-C10	ThunderSync3-C10
Universal	Pala Sub-16	Pala Sub-16
Universal	U8S	U8S
Sincronización PD	PDSync-C4	PDSync-C4
Universal	ModIT-Max	ModIT-Max
Control de motor	Tablero de control de motores	ModIT-Max

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_firmware",
  "parámetros": [ llamada de firmware ]
}
```

llamada de firmware	Descripción
agregar	Proporciona firmware a la API como fuente de actualización disponible. ver cbrx_firmware (añadir)
eliminar	Elimina el firmware para que no esté disponible para la API. ver cbrx_firmware (eliminar)
lista	Muestra todo el firmware disponible. ver cbrx_firmware (lista)
actualizar	Inicie la actualización del firmware. ver cbrx_firmware (actualización)
estado	Obtenga el estado de la actualización del firmware. ver cbrx_firmware (estado)

5.24. cbrx_firmware (añadir)

Agrega el archivo de firmware al almacenamiento local de la API en el host que ejecuta la API. La adición de archivos de firmware se realiza proporcionando un zip codificado en Base64 del archivo. Para obtener más información sobre la codificación Base64, consulte el siguiente [enlace](#).

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_firmware",
  "parámetros": [
    "agregar",
    " nombre de archivo ", bytes codificados
  ]
}
```

Parámetros	Descripción
<i>Nombre del archivo</i>	El nombre del archivo de firmware
<i>bytes codificados</i>	El zip codificado del archivo en Base64

Devoluciones:

```
{
  "resultado": verdadero
}
```

Ejemplos

Ejemplo de solicitud JSON-RPC: (tenga en cuenta que en este ejemplo el texto codificado en base64 se ha reducido para simplificar y aclarar el manual)

```
{
```

```
"jsonrpc": "2.0",  
"id": 0,  
"method": "cbrx_firmware",  
"params": [  
  "add",  
  "CambrionixFirmware-v1.87-un.enfir",  
  "eJwsnduOLTFSbd9Lqn/8f8BywVzYg=="  
]  
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": true  
}
```

5.25. cbrx_firmware (lista)

Obtener una lista de todas las versiones de firmware disponibles

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_firmware",
  "parámetros": ["lista"]
}
```

Devoluciones:

```
{
  "resultado": [
    "nombre de archivo": "nombre de archivo ", "versión": " versión de firmware "
  ]
}
```

Parámetro	Descripción
<i>Nombre del archivo</i>	Nombre del archivo
<i>versión de firmware</i>	Número de versión del firmware

Ejemplo

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_firmware",
  "params": [
    "list"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": [
    {
      "filename": "CambrionxFirmware-1.0.3+39-00-st.enfir",
      "version": "1.0.3+39"
    },
    {
      "filename": "CambrionxFirmware-v1.86-un.enfir",
      "version": "1.86"
    },
    {
      "filename": "CambrionxFirmware-v1.87-un.enfir",
      "version": "1.87"
    }
  ]
}
```

5.26. cbrx_firmware (eliminar)

Elimine el archivo de firmware del almacenamiento local de la API en el host que ejecuta la API.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_firmware",
  "parámetros": [
    "eliminar", "nombre de archivo "
  ]
}
```

filename es el nombre del archivo de firmware.

Devoluciones:

```
{
  "resultado": verdadero
}
```

Ejemplo

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": "0",
  "method": "cbrx_firmware",
  "params": [
    "remove",
    "CambrionxFirmware-v1.86-un.enfir"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": "0",  
  "result": true  
}
```

5.27. cbrx_firmware (estado)

Este método se puede utilizar para obtener el estado de una actualización de firmware.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_firmware",
  "parámetros": [
    "estado",
    " mango de conexión ",
  ]
}
```

manija de conexión es el [Connection Handles](#) como un entero

Devoluciones:

```
{
  "resultado": [
    "Versión": " versión de firmware ", "Tipo": " tipo de firmware ",
    "Progreso": porcentaje de progreso , "Etapa": " valor de etapa " ]
}
```

Parámetro	Descripción
<i>versión de firmware</i>	Número de versión del firmware
<i>tipo de firmware</i>	Se utiliza para indicar el tipo de firmware
<i>porcentaje de progreso</i>	El progreso de la actualización como un porcentaje
<i>valor de etapa</i>	La "etapa" en la que se encuentra actualmente la actualización del firmware

valor de etapa	Descripción
none	El firmware no se actualiza
connecting	Connecting to the hub to update the firmware
en eso	La actualización se está inicializando.
borrando	Borrado del firmware actual
borrado	El firmware actual ha sido borrado
updating	Se está instalando el nuevo firmware
updated	El nuevo firmware ha terminado de instalarse
verifying	Comprobación de que el firmware se ha instalado correctamente
completo	El cheque se ha completado
reiniciando	Reiniciar el concentrador después de completar todas las comprobaciones y la instalación
reiniciado	Hub se ha reiniciado y está listo para usar
skipped	Update skipped as hub already updated

errores

Si hay un error en la etapa, aparecerá uno de los siguientes errores en los valores de la etapa. Cualquiera de estos errores de etapa significa que el firmware del concentrador se encuentra en un estado no válido y se debe volver a realizar.

error de etapa	Descripción
crypt-init-falló	Se utilizó el tipo de firmware incorrecto para el dispositivo seleccionado
init-falló	La etapa de inicialización falló
erase-failed	El firmware actual no se pudo borrar
flash fallido	El nuevo firmware no se pudo instalar en el concentrador
comprobación fallida	Las comprobaciones de instalación fallaron
reinicio fallido	No se pudo reiniciar el concentrador

Ejemplo

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_firmware",
  "params": [
    "status",
    "7654"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": [
    "Version": "1.79",
    "Type": "un",
    "Progress": 60,
    "Stage": "verifying"
  ]
}
```

5.28. cbrx_firmware (actualización)

Utilice este método para iniciar actualizaciones de firmware, algunos productos pueden tener varios firmware, como pantalla, firmware de control de motor, etc. Puede iniciar la actualización de varios firmware con un solo comando como se documenta en el ejemplo.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_firmware",
  "parámetros": [
    "actualizar",
    " mango de conexión ",
    " firmware-type " " filename " ]
}
```

Parámetro	Descripción
<i>tipo de firmware</i>	This is a two letter code to define the specific firmware type on the device, more information on the firmware type can be found in the cbrx_firmware section
<i>Nombre del archivo</i>	Nombre del archivo. Se puede usar un archivo de firmware particular de los disponibles, para obtener el uso de archivos disponibles cbrx_firmware (lista)
<i>empuñadura de conexión</i>	El Connection Handles como un entero

From version 3.9 onwards with the Cambrionix Hub API you can substitute the connection-handle with the hubs serial number.

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API, se devolverá un objeto de error JSON.

Ejemplo

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": "0",
  "method": "cbrx_firmware",
  "params": [
    "update",
    "7654",
    {
      "un": "CambrionxFirmware-v1.86-un.enfir",
      "mc": "CambrionxFirmware-v1.0.0-mc.enfir"
    }
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": "0",
  "result": true
}
```

5.29. cbrx_get_usb (tree)

Devuelve todo el árbol USB que se ha descubierto.

Sintaxis: ver Call Structure

```
{
  "method": "cbrx_get_usb",
  "params": ["tree"]
}
```

Devoluciones

```
{
  "resultado": [
    {
      "VID": ID de proveedor , "PID": ID de producto , "Descripción":
      " descripción " , "LocationID": ID de ubicación , "USBVersion":
      versión USB , "USBPower": {
        "Estado": " estado de potencia " , "Descripción": " descripción
        de potencia " } ,
      "Controlador de host": {
        "Tipo": " tipo de controlador de host " , "EndpointTotal": puntos
        finales activos , "EndpointPeakTotal": puntos finales pico ,
        "EndpointMemoryUsed": memoria endpoint ,
        "EndpointPeakMemoryUsed": memoria de punto final pico
      } ,
      "niños": [
        {
          "VID": ID de proveedor , "PID": ID de producto , "LocationID":
          ID de ubicación , "USBVersion": versión USB , "USBPower":
          {
            "Estado": " estado de potencia " , "Descripción": " descripción
```

```

de potencia " },
"Velocidad USB":
    {
"Velocidad": " velocidad USB ", "Descripción": " nombre-
velocidad " },
"Puntos finales":
    {
"Active": active-endpoints, "Maximum": maximum-endpoints,
"Memory": endpoint-memory,
"TotalInTree": {
"Endpoints": tree-endpoints
"Memory": tree-memory
    }
    }
    }
    ]
    }
    ]
}

```

Producción	Descripción
<i>Id. de proveedor</i>	Device Vendor ID, or VID. Mostrado como un entero
<i>ID del Producto</i>	Product ID, PID. Mostrado como un entero
<i>descripción</i>	nombre del equipo
<i>id-ubicación</i>	El Location IDs como un entero
<i>versión usb</i>	El número de versión USB de la conexión al concentrador. Formato 'N.nn'
<i>estado de poder</i>	USB Power States código
<i>poder-descripción</i>	Alimentación USB encendida/apagada

Producción	Descripción
<i>tipo de controlador de host</i>	El tipo de controlador de host USB
<i>puntos finales máximos</i>	Uso máximo de punto final en el controlador de host USB. ver Puntos finales
<i>pico-punto final-memoria</i>	Uso máximo de memoria de punto final. ver Puntos finales
<i>Velocidad USB</i>	Conexión USB de máxima velocidad capaz de
<i>velocidad-nombre</i>	Nombre de la conexión USB, es decir USB de supervelocidad de 5 Gbps
<i>puntos finales activos</i>	Cuántos puntos finales está usando el dispositivo
<i>puntos finales máximos</i>	Cuántos puntos finales es capaz de usar el dispositivo
<i>memoria de punto final</i>	Cantidad de memoria utilizada por los puntos finales
<i>tree-endpoints</i>	How many endpoints the tree is using
<i>tree-memory</i>	Amount of memory being used by endpoints in the tree

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_get_usb",
  "params": ["tree"]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
```

```

"result": [
  {
    "VID": 32902,
    "PID": 40429,
    "Description": "Intel(R) USB 3.1 eXtensible HostController - 1.10
      (Microsoft)", "LocationID": 553648128, "USBVersion": 3.1, "USBPower": {
      "State": "D0",
      "Description": "On"
    }
  },
  "HostController": {
    "Type": "XHCI",
    "EndpointTotal": 9,
    "EndpointPeakTotal": 60,
    "EndpointMemoryUsed": 57344,
    "EndpointPeakMemoryUsed": 331776
  },
  "children": [
    {
      "VID": 3141,
      "PID": 26403,
      "LocationID": 558891008,
      "USBVersion": 2.01,
      "USBPower": {
        "State": "D0",
        "Description": "On"
      },
      "USBSpeed": {
        "Speed": "480Mbps",
        "Description": "High"
      },
      "Endpoints": {
        "Active": 2,
        "Maximum": 3,
        "Memory": 12288
      }
    },
    {
      "VID": 1161,
      "PID": 57506,
      "LocationID": 560988160,
      "USBVersion": 1.1,
      "USBPower": {
        "State": "D2",
        "Description": "Low power"
      },
      "USBSpeed": {
        "Speed": "12Mbps",
        "Description": "Full"
      },
      "Endpoints": {
        "Active": 6,
        "Memory": 24576
      }
    }
  ],
  {

```

```
"VID": 0,  
"PID": 0,  
"LocationID": 563085312,  
"USBVersion": 0,  
"USBSpeed": {  
  "Speed": "1.5Mbps",  
  "Description": "Low"  
},  
"Endpoints": {  
  "Active": 1,  
  "Memory": 4096  
}  
]  
}  
]
```

5.30. cbrx_get_usb (descriptors)

Request entire dump of a USB device's descriptor information. This can be a lot of data for some devices (especially phones and tablets).

Sintaxis: ver Call Structure

```
{
  "method": "cbrx_get_usb",
  "params": ["descriptors", locationID | hub-serial "]
}
```

Variable	Descripción
<i>locationID</i>	El Location IDs como un entero
<i>hub-serial</i>	Este es el número de serie del hub devuelto de cbrx_descubrir

Devoluciones

All variable names in the returned data match the names in [Chapter 9 of the USB 3.2 specification](#) for ease of reference. Each descriptors raw fields (as taken from the USB 3.2 spec) are represented first, and where appropriate an additional "Derived" member will be present that shows bitfields or resolved string descriptors.

For example, on the device's main descriptor there is an iManufacturer field, which is the index of the string descriptor used for that name. This will also be present in Derived.Manufacturer.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "method": "cbrx_get_usb",
  "params": ["descriptors", "123456789abcdef"],
  "id": 0
}
```

Example successful response from a standard USB flash drive:

```

{
  "jsonrpc": "2.0",
  "id": 0,
  "result": {
    "RawBytes": "120120030000000951092b17010001020301",
    "bLength": 18,
    "bDescriptorType": 1,
    "bNumConfigurations": 1,
    "bcdUSB": 800,
    "bDeviceClass": 0,
    "bDeviceSubClass": 0,
    "bDeviceProtocol": 0,
    "bMaxPacketSize0": 9,
    "idVendor": 2385,
    "idProduct": 5931,
    "bcdDevice": 1,
    "iManufacturer": 1,
    "iProduct": 2,
    "iSerialNumber": 3,
    "Derived": {
      "DescriptorType": "Device",
      "CurrentConfiguration": 1,
      "DeviceClass": "Reserved"
    },
    "Configurations": {
      "1": [
        {
          "RawBytes": "09022c00010100ff8025",
          "bLength": 9,
          "bDescriptorType": 2,
          "wTotalLength": 44,
          "bConfigurationValue": 1,
          "bmAttributes": 128,
          "bNumDescriptors": 1,
          "iConfiguration": 0,
          "reserved1": 0,
          "reserved2": 1,
          "SelfPowered": 0,
          "Derived": {
            "DescriptorType": "Configuration",
            "MaxPower": 37,
            "RemoteWakeUp": 0,
            "BusPowered": 0
          }
        },
        {
          "RawBytes": "090400000208065000",
          "bLength": 9,
          "bDescriptorType": 4,
          "iInterface": 0,
          "bInterfaceNumber": 0,
          "bAlternateSetting": 0,
          "bNumEndpoints": 2,
          "bInterfaceClass": 8,
          "bInterfaceSubClass": 6,

```

```

    "bInterfaceProtocol": 80,
    "Derived": {
      "DescriptorType": "Interface"
    }
  },
  {
    "RawBytes": "0705ff8102000400",
    "bLength": 7,
    "bDescriptorType": 5,
    "bInterval": 0,
    "bEndpointAddress": 129,
    "bmAttributes": 2,
    "wMaxPacketSize": 1024,
    "Derived": {
      "DescriptorType": "Endpoint",
      "EndpointAddress": 1,
      "Direction": "In",
      "Type": "Bulk"
    }
  },
  {
    "RawBytes": "06300f000000",
    "bLength": 6,
    "bDescriptorType": 48,
    "wBytesPerInterval": 0,
    "bMaxBurst": 15,
    "bmAttributes": 0,
    "Derived": {
      "DescriptorType": "SuperSpeedEndpointCompanion",
      "MaxStreams": 0,
      "Mult": 0,
      "SspCompanion": 0
    }
  },
  {
    "RawBytes": "07050202000400",
    "bLength": 7,
    "bDescriptorType": 5,
    "bInterval": 0,
    "bEndpointAddress": 2,
    "bmAttributes": 2,
    "wMaxPacketSize": 1024,
    "Derived": {
      "DescriptorType": "Endpoint",
      "EndpointAddress": 2,
      "Direction": "Out",
      "Type": "Bulk"
    }
  },
  {
    "RawBytes": "06300f000000",
    "bLength": 6,
    "bDescriptorType": 48,
    "wBytesPerInterval": 0,
    "bMaxBurst": 15,

```

```

        "bmAttributes": 0,
        "Derived": {
            "DescriptorType": "SuperSpeedEndpointCompanion",
            "MaxStreams": 0,
            "Mult": 0,
            "SspCompanion": 0
        }
    }
}
},
"Strings": {
    "1": "Kingston",
    "2": "DataTraveler 70",
    "3": "1831BFBD3065F551C96001E7"
},
"BOS": {
    "RawBytes": "050f160002",
    "bLength": 5,
    "bDescriptorType": 15,
    "Derived": {
        "DescriptorType": "BOS"
    },
    "wTotalLength": 22,
    "bNumDescriptors": 2,
    "Capabilities": [
        {
            "RawBytes": "07100206000000",
            "bLength": 7,
            "bDescriptorType": 16,
            "bDevCapabilityType": 2,
            "bmAttributes": 6,
            "Derived": {
                "DescriptorType": "DeviceCapability",
                "CapabilityType": "USB20Extension",
                "LPMCapable": 1,
                "BESLAndAlternateHIRDSupported": 1,
                "BaselineBESLValid": 0,
                "DeepBESLValid": 0,
                "BaselineBESL": 0,
                "DeepBESL": 0
            }
        },
        {
            "RawBytes": "0a1003000e00020affff07",
            "bLength": 10,
            "bDescriptorType": 16,
            "bDevCapabilityType": 3,
            "wU2DevExitLat": 2047,
            "bmAttributes": 0,
            "wSpeedsSupported": 14,
            "bFunctionalitySupport": 2,
            "bU1DevExitLat": 10,
            "Derived": {
                "DescriptorType": "DeviceCapability",
                "CapabilityType": "SuperSpeedUSB",

```

```
    "LTMCapable": 0,  
    "SpeedsSupported": [  
      "Full",  
      "High",  
      "SuperSpeed"  
    ]  
  }  
}  
]  
}  
}''''
```

5.31. cbrx_hub_get

Desde el concentrador especificado por el número de serie de los concentradores, obtenga el valor de la clave. Similar a [cbrx_conexión_get](#) .

Tenga en cuenta que esta función será más lenta si necesita realizar varias operaciones en el mismo concentrador.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    hub-serial, dictionary-key
  ]
}
```

Parámetro	Descripción
<i>hub-serial</i>	Este es el número de serie del hub devuelto de cbrx_descubrir
<i>clave-diccionario</i>	Ver Obtener Diccionario para más información

Devoluciones:

```
{
  "resultado": [ valor-diccionario ] }
```

valor-diccionario es el valor clave que se especifica ver [Obtener Diccionario](#) para más información.

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_hub_get",
  "params": [
    "000000897FD0505A",
    "nrOfPorts"
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 15
}
```

5.32. cbrx_hub_set

En el concentrador especificado por el identificador de conexión, establezca el valor de la clave. Similar a [conjunto_conexión_cbrx](#)

Tenga en cuenta que esta función será más lenta si necesita realizar varias operaciones en el mismo concentrador.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_hub_set",
  "parámetros": [
    hub-serial, dictionary-key, Value
  ]
}
```

Parámetro	Descripción
<i>hub-serial</i>	Este es el número de serie del hub devuelto de cbrx_descubrir
<i>clave-diccionario</i>	Ver Establecer diccionario para más información
<i>Valor</i>	El valor que desea aplicar a la clave

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_hub_set",
  "params": [
    "7FD0505A",
    "TwelveVoltRail.OverVoltage",
    true
  ]
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": "0",
  "result": true
}
```

5.33. cbrx_notificaciones

La API admite el envío de notificaciones para ciertos eventos. Los paquetes de notificación son iguales a otros objetos de respuesta JSON-RPC, excepto que no tienen un campo "id".

Nota: Las notificaciones solo se envían a las conexiones de socket activas que las han solicitado. Cerrar un socket y abrir otro significará que debe volver a solicitar notificaciones. Ver el [Inicio rápido](#) por ejemplo

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_notificaciones",
  "parámetros": [" notificación "], }

```

la *notificación* es de la matriz de cadenas que se muestra al enviar [cbrx_apidetalles](#) una lista completa de posibles notificaciones se puede encontrar en la sección [Notificaciones de la API](#)

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_notificaciones",
  "params": [

```

```
    "usb-device-attached"  
  ]  
}
```

Ejemplo de respuesta exitosa:

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": true  
}
```

5.34. cbrx_pair_device

Initiate pairing of an iOS device. This is not usually necessary as it will occur automatically when the API attempts to query battery information.

Sintaxis: ver API Call Structure

```
{
  "method": "cbrx_pair_device",
  "params": ["UDID"]
}
```

Variable	Descripción
<i>UDID</i>	The device USB serial number

Devoluciones

```
{
  "resultado": verdadero
}
```

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_pair_device",
  "params": ["1234567"]
}
```

Ejemplo de respuesta exitosa:

```
{
```

```
"jsonrpc": "2.0",  
"id": 0,  
"result": true  
}
```

6. Notificaciones de la API

Los paquetes de notificación API son iguales a otros paquetes JSON-RPC, excepto que no tienen un campo de "id". La mayoría de estas notificaciones no proporcionan nada en el campo "parámetros". Estas notificaciones solo se enviarán si están habilitadas mediante el [cbrx_notificaciones](#) método.

Puede encontrar una lista de posibles notificaciones disponibles utilizando nuestra API en la siguiente tabla. La mayoría de estas notificaciones no proporcionan nada en el campo "parámetros".

Notificación	Descripción
todo	Solicitar todas las notificaciones
descubrir-cambiado	La API detectó un cambio en los concentradores disponibles. Deberías volver a ejecutar cbrx_descubrir en este punto.
muerto-hub-cambiado	La API detectó que un concentrador dejó de responder o no se puede conectar a
firmware-progreso	Solicitar actualizaciones sobre el progreso de la actualización del firmware
exceso de temperatura	Hub está sobre la temperatura
sobretensión	Hub tiene sobrevoltaje
rfid-recibido	Reciba una notificación cuando se presente una tarjeta RFID a un sensor
rfid-removido	Reciba una notificación cuando se extraiga una tarjeta RFID de un sensor
dispositivo usb conectado	Más detallado que usb-changed , esto le indicará los dispositivos específicos que se han conectado.
dispositivo usb desconectado	Más detallado que usb-changed , esto le indicará los dispositivos específicos que se han desconectado.
bajo voltaje	El concentrador está bajo voltaje
usb-cambiado	La API detectó un cambio en el árbol USB.

descubrir-cambiado

Se ha detectado un cambio en los concentradores que están disponibles para la API.

Paquete de notificación de ejemplo:

```
{
  "jsonrpc": "2.0",
  "method": "discover-changed"
}
```

muerto-hub-cambiado

La API detectó que un concentrador dejó de responder o no se puede conectar, por ejemplo, porque otro programa tiene su puerto serie abierto.

Paquete de notificación de ejemplo:

```
{
  "jsonrpc": "2.0",
  "method": "dead-hub-changed",
  "params": {
    "IsDead": True
  }
}
```

firmware-progreso

Información sobre el progreso de la actualización del firmware. Para obtener más información sobre la salida, consulte [cbrx_firmware \(estado\)](#)

Paquete de notificación de ejemplo:

```
{
  "jsonrpc": "2.0",
  "method": "firmware-progress",
  "params": {
    "Progress": 60,
    "Stage": "flashing",
    "Type": "charger",
    "HostDevice": "1212343456567878",
    "HostSerial": "/dev/tty.usbmodem1421502",
    "HostDescription": "PS15-USB3"
  }
}
```

exceso de temperatura

El concentrador está por encima de la temperatura de funcionamiento; consulte [los manuales de usuario del producto](#) para obtener más detalles.

Paquete de notificación de ejemplo:

```
{
  "jsonrpc": "2.0",
  "method": "over-temperature"
}
```

sobretensión

El concentrador tiene un voltaje superior al recomendado; consulte [los manuales de usuario del producto](#) para obtener más detalles.

Paquete de notificación de ejemplo:

```
{
  "jsonrpc": "2.0",
  "method": "over-voltage"
}
```

rfid-recibido

Un sensor RFID ha detectado que hay una tarjeta RFID presente.

Paquete de notificación de ejemplo:

```
{
  "jsonrpc": "2.0",
  "method": "rfid-received",
  "params": "12784556655489628"
}
```

rfid-removido

Un sensor RFID ha detectado que se ha retirado una tarjeta RFID.

Paquete de notificación de ejemplo:

```
{
  "jsonrpc": "2.0",
  "method": "rfid-removed",
  "params": "12784556655489628"
}
```

dispositivo usb conectado

Un dispositivo está disponible para la API y se presenta una salida de información detallada sobre el dispositivo.

Paquete de notificación de ejemplo:

```
{
  "jsonrpc": "2.0",
  "method": "usb-device-attached",
  "params": {
    "HostDevice": "1212343456567878",
    "HostSerial": "/dev/tty.usbmodem1421502",
    "HostPort": 7,
    "HostDescription": "PS15-USB3",
    "USB2": {
      "Description": "iPhone",
      "LocationID": 573710336,
      "Manufacturer": "Apple Inc.",
      "PID": 4776,
      "SerialNumber": "012a37d1fa07617ad7ef0430ba49f479ab9fb6b8",
      "USBVersion": 2,
      "VID": 1452
    }
  }
}
```

dispositivo usb desconectado

Un dispositivo ya no está disponible para la API y se presenta una salida de información detallada sobre el dispositivo.

Paquete de notificación de ejemplo:

```
{
  "jsonrpc": "2.0",
  "method": "usb-device-detached",
  "params": {
    "HostDevice": "1212343456567878",
    "HostSerial": "/dev/tty.usbmodem1421502",
    "HostPort": 7,
    "HostDescription": "PS15-USB3",
    "USB2": {
      "Description": "iPhone",
      "LocationID": 573710336,
      "Manufacturer": "Apple Inc.",
      "PID": 4776,
      "SerialNumber": "012a37d1fa07617ad7ef0430ba49f479ab9fb6b8",
      "USBVersion": 2,
      "VID": 1452
    }
  }
}
```

bajo voltaje

El concentrador tiene el voltaje recomendado; consulte [los manuales de usuario del producto](#) para obtener más detalles.

Paquete de notificación de ejemplo:

```
{
  "jsonrpc": "2.0",
  "method": "under-voltage"
}
```

usb-cambiado

Ha habido un cambio en el árbol USB.

Paquete de notificación de ejemplo:

```
{  
  "jsonrpc": "2.0",  
  "method": "usb-changed"  
}
```

7. Métodos obsoletos

Estos métodos existen solo para admitir la compatibilidad con versiones anteriores y no deben usarse. Estos métodos pueden eliminarse en futuras versiones.

Llamada API	Descripción
cbrx_apiversion (verdadero)	Obtener una versión detallada de la API

7.1. cbrx_apiversion (verdadero)

!Este método quedó obsoleto en la versión 3.0 de la API, utilice 'cbrx_apidetalles'!

Devuelve una versión detallada de la API en ejecución.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_apiversion",
  "parametros": [verdadero]
}
```

Devoluciones:

```
{
  "resultado": {
    "versión": [ número de versión ], "semver": " variante de semver ",
    "commitid": número de confirmación , "sucursal": " nombre de sucursal ",
    "capacidad": [ capacidad de API ], "notificaciones": [ notificación posible ],
    "instalación": " ubicación de instalación ", "registro": " ubicación de registros ",
    "configuración": " ubicación de configuración ",
    "documentación": " ubicación de documentación ", "cpu ": {
      "marca": " información de marca ", "arco": " arquitectura de CPU ",
      "características": [ características de CPU ], "núcleos": valor de núcleos
    },
    "os": " Información del SO " }
}
```

Producción	Descripción
<i>número de versión</i>	Número de versión de la API como un número entero (Mayor, Menor, Revisión, Compilación)

Producción	Descripción
<i>variante semver</i>	El nombre completo de la versión de la API
<i>variante-commitid</i>	El valor numérico del ID de confirmación
<i>nombre-sucursal</i>	La rama de la API instalada
<i>API-capacidad</i>	Información API para uso interno de Cambrionix
<i>notificación posible</i>	Matriz de cadenas para mostrar posibles notificaciones. ver Notificaciones de la API
<i>ubicación de instalación</i>	La ubicación de los archivos de instalación
<i>ubicación de registros</i>	La ubicación donde se almacenan los registros
<i>configuración-ubicación</i>	La ubicación de la configuración de la API
<i>documentación-ubicación</i>	La dirección web de la documentación de la API
<i>información de marca</i>	La marca de la CPU
<i>CPU-arquitectura</i>	La arquitectura de la CPU
<i>Características de la CPU</i>	Funciones disponibles en la CPU
<i>valor de núcleos</i>	Cuántos cores tiene la cpu
<i>os-información</i>	Sistema operativo que se ejecuta en la máquina local

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_apiversion",
  "params": [
    True
  ]
}
```

Ejemplo de respuesta exitosa

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": {
    "version": [
      3,
      7,
      0,
      34
    ],
    "semver": "3.7.0+34",
    "guid": {
      "id": "d0dc3cac-e165-4e38-88bb-39064431bdc9",
      "computerId": "35aea4bc-44d3-4e9e-9b3c-c33b965c5639"
    },
    "host": [
      {
        "ip": "10.167.111.81",
        "port": 0,
        "nameServer": "10.167.111.241",
        "domainName": "CBRX.LOCAL",
        "hostName": "CBRXPC-011",
        "adapterName": "Intel(R) Ethernet Controller (3) I225-V",
        "adapterType": "Ethernet"
      }
    ],
    "commitid": 4287981321,
    "branch": "release",
    "capability": [
      "protobuf",
      "crash-report",
      "notification"
    ],
    "notifications": [
      "usb-changed",
      "usb-device-attached",
      "usb-device-detached",
      "discover-changed",
      "dead-hub-changed",
      "firmware-progress",
      "rfid-received",
    ]
  }
}
```

```

    "rfid-removed",
    "over-temperature",
    "over-voltage",
    "under-voltage",
    "certificate-changed"
  ],
  "install": "C:\\Program Files\\Cambrionix\\API",
  "logging": "C:\\ProgramData\\Cambrionix\\Log",
  "settings": "C:\\ProgramData\\Cambrionix",
  "documentation": "C:\\Program Files\\Cambrionix\\API\\Cambrionix Hub API
Reference.html",
  "cpu": {
    "brand": "12th Gen Intel(R) Core(TM) i9-12900K",
    "arch": "x64",
    "features": [
      "aes",
      "avx",
      "avx2",
      "bmi1",
      "bmi2",
      "clflushopt",
      "clflush",
      "clwb",
      "cx16",
      "cx8",
      "erms",
      "f16c",
      "fma3",
      "fpu",
      "mmx",
      "movbe",
      "pclmulqdq",
      "popcnt",
      "rdrnd",
      "rdseed",
      "sha",
      "smx",
      "ss",
      "sse",
      "sse2",
      "sse3",
      "sse4_1",
      "sse4_2",
      "ssse3",
      "tsc",
      "vaes",
      "vpclmulqdq"
    ],
    "cores": 24
  },
  "os": "Windows 10 Pro 21H2 Build 19044.1889 64-bit"
}

```

7.2. cbrx_get_usbtree

This method was deprecated in API version 3.5 please use 'cbrx_get_usb (tree)'

Devuelve todo el árbol USB que se ha descubierto.

Sintaxis: ver API Call Structure

```
{
  "método": "cbrx_get_usbtree",
}
```

Devoluciones

```
{
  "resultado": [
    {
      "VID": ID de proveedor , "PID": ID de producto , "Descripción":
      " descripción " , "LocationID": ID de ubicación , "USBVersion":
      versión USB , "USBPower": {
      "Estado": " estado de potencia " , "Descripción": " descripción
      de potencia " } ,
      "Controlador de host": {
      "Tipo": " tipo de controlador de host " , "EndpointTotal": puntos
      finales activos , "EndpointPeakTotal": puntos finales pico ,
      "EndpointMemoryUsed": memoria endpoint ,
      "EndpointPeakMemoryUsed": memoria de punto final pico
      } ,
      "niños": [
        {
          "VID": ID de proveedor , "PID": ID de producto , "LocationID":
          ID de ubicación , "USBVersion": versión USB , "USBPower":
          {
```

```

"Estado": " estado de potencia ", "Descripción": " descripción
de potencia " },
"Velocidad USB":
    {
"Velocidad": " velocidad USB ", "Descripción": " nombre-
velocidad " },
"Puntos finales":
    {
"Activo": puntos finales activos , "Máximo": puntos finales
máximos , "Memoria": memoria de puntos finales
    }
    }
]
}
]
}

```

Producción	Descripción
<i>Id. de proveedor</i>	Device Vendor ID, VID. Mostrado como un entero
<i>ID del Producto</i>	Product ID, PID. Mostrado como un entero
<i>descripción</i>	nombre del equipo
<i>id-ubicación</i>	El Location IDs como un entero
<i>versión usb</i>	El número de versión USB de la conexión al concentrador. Formato 'N.nn'
<i>estado de poder</i>	USB Power States código
<i>poder-descripción</i>	Alimentación USB encendida/apagada
<i>tipo de controlador de host</i>	El tipo de controlador de host USB

Producción	Descripción
<i>puntos finales máximos</i>	Uso máximo de punto final en el controlador de host USB. ver Puntos finales
<i>pico-punto final-memoria</i>	Uso máximo de memoria de punto final. ver Puntos finales
<i>Velocidad USB</i>	Conexión USB de máxima velocidad capaz de
<i>velocidad-nombre</i>	Nombre de la conexión USB, es decir USB de supervelocidad de 5 Gbps
<i>puntos finales activos</i>	Cuántos puntos finales está usando el dispositivo
<i>puntos finales máximos</i>	Cuántos puntos finales es capaz de usar el dispositivo
<i>memoria de punto final</i>	Cantidad de memoria utilizada por los puntos finales

Ejemplos

Ejemplo de solicitud JSON-RPC:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_get_usbtree"
}
```

Ejemplo de respuesta exitosa:

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": [
    {
      "VID": 32902,
      "PID": 40429,
      "Description": "Intel(R) USB 3.1 eXtensible HostController - 1.10 (Microsoft)",
      "LocationID": 553648128,
      "USBVersion": 3.1,
      "USBPower": {
        "State": "D0",
        "Description": "On"
      }
    }
  ],
}
```

```

"HostController": {
  "Type": "XHCI",
  "EndpointTotal": 9,
  "EndpointPeakTotal": 60,
  "EndpointMemoryUsed": 57344,
  "EndpointPeakMemoryUsed": 331776
},
"children": [
  {
    "VID": 3141,
    "PID": 26403,
    "LocationID": 558891008,
    "USBVersion": 2.01,
    "USBPower": {
      "State": "D0",
      "Description": "On"
    },
    "USBSpeed": {
      "Speed": "480Mbps",
      "Description": "High"
    },
    "Endpoints": {
      "Active": 2,
      "Maximum": 3,
      "Memory": 12288
    }
  },
  {
    "VID": 1161,
    "PID": 57506,
    "LocationID": 560988160,
    "USBVersion": 1.1,
    "USBPower": {
      "State": "D2",
      "Description": "Low power"
    },
    "USBSpeed": {
      "Speed": "12Mbps",
      "Description": "Full"
    },
    "Endpoints": {
      "Active": 6,
      "Memory": 24576
    }
  },
  {
    "VID": 0,
    "PID": 0,
    "LocationID": 563085312,
    "USBVersion": 0,
    "USBSpeed": {
      "Speed": "1.5Mbps",
      "Description": "Low"
    },
    "Endpoints": {

```

```
        "Active": 1,  
        "Memory": 4096  
    }  
  ]  
}  
]
```

8. Cadena de dispositivo

Cuando la API consulta un dispositivo, puede devolver la siguiente cadena.

```

"Dispositivo": {
  "VID": vendor-ID,
  "PID": ID de producto ,
  "Fabricante": " fabricante del dispositivo ",
  "Descripción": " descripción ",
  "SerialNumber": "usb-serial",
  "LocationID": ubicación-id ,
  "DevicePath": device-path
  "USBVersion": versión usb ,
  "Alimentación USB": {
    "Estado": " estado de poder ",
    "Descripción": " descripción de poder "
      },
  "Velocidad USB": {
    "Velocidad": " velocidad USB ",
    "Descripción": "Descripción USB "
    "Capacidad": {
      "Velocidad": " velocidad capaz ",
      "Descripción": " descripción capaz "
        }
      },
  "Puntos finales": {
    "Activo": puntos finales activos ,
    "Máximo": puntos finales máximos ,
    "Memoria": memoria de punto final
  }
}

```

```
    },  
    "Batería": {  
        "DataSource": " batería-fuente-de-datos ",  
        "TrustLevel": " nivel de confianza ",  
        "PairingSupported": emparejamiento compatible ,  
        "CurrentLevel": nivel de corriente de la batería ,  
        "HoraActual": hora-del-centro-actual ,  
        "StartingLevel": nivel de inicio de carga ,  
        "StartingTime": tiempo de inicio de carga ,  
        "CapacityNew": nueva-batería-capacidad ,  
        "Capacidad": capacidad actual de la batería ,  
        "ChargingStatus": " estado de carga ",  
        "HealthPercent": battery-health,  
        "Temperature": device-temperature  
    },  
    "PhoneSerialNumber": " teléfono-serie ",  
    "PhoneIdentity": " nombre-teléfono ",  
    "PhoneModel": "phone-model",  
    "IMEI": " Número IMEI ",  
    "MacAddress": " MacAddress ",  
    "PhoneSoftwareVersion": "phone-OS-version",  
    "PhoneECID": ECID,  
    "PhoneProductType": "phone-product",  
    "PhoneOSType": "phone-OS",  
    "PhoneColour": "phone-colour"  
    }  
}
```

Producción	Descripción
<i>vendor-ID</i>	Device Vendor ID number, VID. Mostrado como un entero
<i>ID del Producto</i>	Product ID number, PID. Mostrado como un entero
<i>fabricante del dispositivo</i>	El nombre del fabricante del dispositivo.
<i>descripción</i>	Nombre del equipo
<i>usb-serial</i>	USB serial number
<i>id-ubicación</i>	El Location IDs como un entero
<i>device-path</i>	The platform specific path for the device
<i>versión usb</i>	The USB version number of the connection to the hub
<i>estado de poder</i>	USB Power States código
<i>poder-descripción</i>	Alimentación USB encendida/apagada
<i>Velocidad USB</i>	Conexión USB de máxima velocidad capaz de
<i>USB-descripción</i>	Nombre de la conexión USB, es decir USB de supervelocidad de 5 Gbps
<i>velocidad capaz</i>	Dispositivo de máxima velocidad de datos capaz de
<i>descripción capaz</i>	Nombre del dispositivo de máxima velocidad de datos capaz de
<i>puntos finales activos</i>	How many endpoints the device is using, displayed as a integer
<i>puntos finales máximos</i>	How many endpoints the device is capable of using, displayed as a integer
<i>memoria de punto final</i>	Amount of memory being used by endpoints, displayed as a integer
<i>batería-fuente-de-datos</i>	La fuente de la información de la batería del dispositivo
<i>nivel de confianza</i>	Si el dispositivo es de confianza/emparejado

Producción	Descripción
<i>emparejamiento de apoyo</i>	Whether the device supports being trusted/ paired
<i>nivel de corriente de la batería</i>	Current battery percent level of device, displayed as a integer
<i>tiempo-del-centro-actual</i>	The hub time in ms, shown as an integer in ms
<i>nivel de inicio de carga</i>	Battery percentage level when device connected, displayed as a integer
<i>tiempo de inicio de carga</i>	The hub time in ms charging started, shown as an integer in ms
<i>nueva-bateria-capacidad</i>	The battery capacity of device from new, displayed as a integer
<i>capacidad actual de la batería</i>	The battery capacity of the device now, displayed as a integer
<i>estado de carga</i>	Estado de carga de la batería, es decir lleno
<i>salud de la batería</i>	Battery health percentage, displayed as a integer
<i>device-temperature</i>	The temperature the device is reporting
<i>teléfono-serie</i>	Número de serie del teléfono
<i>nombre de teléfono</i>	Nombre del teléfono
<i>phone-model</i>	The model of the phone i.e. "iPhone 12"
<i>número IMEI</i>	The IMEI number of the phone
<i>Dirección MAC</i>	Una dirección única asignada al dispositivo móvil. Es un valor de 48 bits, que consta de doce caracteres hexadecimales
<i>versión del sistema operativo del teléfono</i>	Version number of the OS on the phone
<i>ECID</i>	Exclusive Chip Identification also referred to as Unique Chip ID

Producción	Descripción
<i>phone-product</i>	Mobile device code i.e. "iPhone13,2"
<i>phone-OS</i>	The OS version of the phone i.e. "iPhone OS"
<i>phone-colour</i>	The colour of the phone

9. API Management

9.1. Stopping the API service

To stop the Cambrionix Hub API service, the process varies slightly depending on your operating system. Below are detailed instructions for Windows, macOS and Linux users.

ventanas

If you wish to stop the Cambrionix Hub API service on a Windows machine, you can easily do so through the Task Manager:

1. Open Task Manager:

Right-click on the Taskbar and select Task Manager, or press Ctrl + Shift + Esc to open it directly.

2. Navigate to Services:

In Task Manager, click on the Services tab to view all the services running on your system.

3. Locate the 'CambrionixApiService' Service:

Scroll through the list of services until you find 'CambrionixApiService'.

4. Stop the Service:

Right-click on the 'CambrionixApiService' service and select Stop from the context menu. This will immediately stop the service, halting all Cambrionix Hub API related functionality until the service is restarted.

linux

For Linux, the Cambrionix Hub API service can be stopped from the command line. Most modern Linux distributions, such as Ubuntu, Fedora, and Debian, use systemd for service management.

To stop the service, use the following command:

```
sudo systemctl stop CambrionixApiService
```

Mac OS

On macOS, services like the Cambrionix Hub API are managed by launchd, which handles system-wide and user-level services.

To stop the Cambrionix Hub API service, use the following command:

```
sudo /usr/bin/CambrionixApiService--remove
```

9.2. Starting the API Service

To start the Cambrionix Hub API service, the process will vary depending on your operating system. Below are the instructions for Windows, Linux, and macOS users.

ventanas

To start the Cambrionix Hub API service on a Windows machine, follow these steps:

1. Open Task Manager:

Right-click on the Taskbar and select Task Manager, or press Ctrl + Shift + Esc to open it directly.

2. Navigate to Services:

In Task Manager, click on the Services tab to view all the services running on your system.

3. Locate the 'CambrionixApiService' Service:

Scroll through the list of services until you find 'CambrionixApiService'.

4. Start the Service:

Right-click on the 'CambrionixApiService' service and select Start from the context menu. This will immediately start the service, enabling the API to handle requests again.

linux

For Linux, where the Cambrionix Hub API service is managed by systemd, you can start the service using the following command:

```
sudo systemctl start CambrionixApiService
```

To check that the service is running, you can use:

```
sudo systemctl status CambrionixApiService
```

Mac OS

On macOS, the Cambrionix Hub API service is managed by launchd. To start the service, use the following command:

```
sudo /usr/bin/CambrionixApiService--install
```

To check if the service is running, you can run:

```
sudo launchctl list | grep cambrionix
```

10. Información adicional

Cambrionix Connect Recorder Service

El servicio de grabadora es un componente de instalación opcional que puede registrar eventos como el estado del dispositivo, el historial de carga y los eventos de conexión. Estos se pueden ver posteriormente en el software del cliente.

Stopping the API service

- ventanas

If you wish to stop the Cambrionix Hub API service from running then open Task Manager, then click through to services and right click the 'CambrionixAPIService' and click 'Stop'

- Linux and macOS

If you wish to stop the Cambrionix Hub API service from running then issue the following command

```
sudo /usr/bin/CambrionixApiService --remove
```

Starting the API Service

- ventanas

To start the Cambrionix Hub API service from running then open Task Manager, then click through to services and right click the 'CambrionixAPIService' and click 'Start'

- Linux and macOS

If you wish to start the Cambrionix Hub API service from running then issue the following command

```
sudo /usr/bin/CambrionixApiService --install
```

Limitaciones

La API proporciona un medio para controlar la mayoría de las funciones de los productos Cambrionix, sin embargo, existen algunas limitaciones. The API can only be used with the following products.

Firmware	Número de parte	nombre del producto
Universal	PP15S	PowerPad15S
Universal	PP15C	PowerPad15C
Universal	PP8S	PowerPad8S
Universal	SS15	SuperSync15
Universal	TS3-16	ThunderSync3-16
ELEGANTE	TS3-C10	ThunderSync3-C10
Universal	Pala Sub-16	Pala Sub-16
Universal	U8S	U8S
Sincronización PD	PDSync-C4	PDSync-C4
Universal	ModIT-Max	ModIT-Max
Control de motor	Tablero de control de motores	ModIT-Max

Preventing Windows from Assigning New COM Ports to Identical USB Devices

When you connect multiple USB devices with the same hardware to a Windows PC, the operating system typically identifies each device by its unique hardware serial number. Windows uses this serial number to assign a new COM port for each device, even if they are identical in terms of hardware. Over time, this can result in a long list of COM ports being assigned, which clutters the device manager and can make managing devices more difficult and confusing.

To prevent this, you can instruct Windows to ignore the hardware serial number for specific USB devices. By doing so, Windows will treat all devices with the same Vendor ID (VID) and Product ID (PID) as a single device and allocate only one COM port, no matter how many of these identical devices are connected over time. This stops the COM port list from unnecessarily filling up.

The following registry entry ensures that Windows ignores the hardware serial number for a USB device with the specific VID 0403 and PID 6015:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\usbflags]
"IgnoreHWSerNum04036015"=hex:01
```

- IgnoreHWSerNum04036015: This entry instructs Windows to ignore the hardware serial number for devices with a Vendor ID (0403) and Product ID (6015). These IDs refer to a specific USB device model.
- hex:01: This value enables the behaviour, telling Windows to treat all devices with this VID and PID as if they have the same serial number.

With this registry setting applied, Windows will no longer assign a new COM port each time an identical device (with the same VID and PID) is connected. Instead, it will reuse the same COM port, preventing the list of COM ports from unnecessarily expanding. This makes it easier to manage USB devices that connect through COM ports, particularly in environments where multiple identical devices are frequently connected and disconnected. It prevents the clutter and confusion caused by having a large number of assigned but unused COM ports.

This solution is particularly useful in scenarios such as:

- Development environments where multiple USB hubs are connected and disconnected.
- Production setups where identical devices are frequently used, and preventing an accumulation of redundant COM ports is necessary.
- Device testing environments where serial communication is crucial, but the aim is to avoid an ever-expanding list of COM ports

Connection Handles

Un identificador de conexión es una representación de una conexión física. El identificador de conexión define no solo qué controlador usar, sino también qué fuente de datos usar con ese controlador. Dentro de un segmento de código, el identificador de conexión identifica una estructura que contiene información de conexión.

Puede obtener un identificador de conexión utilizando [cbrx_conexión_abierta](#)

Si no hay actividad en un identificador abierto durante más de 30 segundos, el identificador dejará de ser válido. Las llamadas posteriores que intenten usar un identificador eliminado fallarán con CBRXAPI_ERRORCODE_INVALIDHANDLE. El software que utiliza la API debe poder hacer frente a esta situación y responder en consecuencia. El software puede simplemente llamar a `cbrx_connection_open` nuevamente para obtener un identificador nuevo, o si se registra para recibir notificaciones, se dejarán abiertas indefinidamente.

Location IDs

Una ID de ubicación es un número entero sin signo de 32 bits que representa la ubicación del dispositivo en el árbol USB. Por ejemplo, si el controlador crea una cadena de ubicación para el dispositivo con el formato `a&b&c`, el ID de ubicación será `0x00000abc`.

Puerto serial

El puerto serie al que está conectado el dispositivo variará en su descripción según el sistema operativo en el que se esté ejecutando la API. Para macOS y Linux, esto se mostrará como la ubicación, para Windows, esto se mostrará como un puerto COM, consulte los ejemplos a continuación.

ventanas

```
COM5
```

Mac OS

```
dev/tty.usbmodem141502
```

linux

```
/dev/ttyUSB0
```

Puntos finales

Si experimenta el límite del punto final, es probable que vea un error que indica que "se quedaron sin recursos USB" en su computadora host, aunque es posible que no vea ningún mensaje de error y (cualquiera de) sus dispositivos USB pueden fallar aleatoriamente funcione correctamente o se vuelva intermitente.

La limitación del extremo USB se aplica a los controladores de host xHCI (USB3) solo en placas base recientes. Estos controladores de host USB tienen una memoria limitada y, por lo general, ofrecen entre 64 y 128 terminales USB.

El problema con la cantidad limitada de puntos finales proporcionados por los controladores de host xHCI es que la mayoría de los dispositivos USB, especialmente los teléfonos móviles y las tabletas, enumeran más de un punto final. Un dispositivo móvil típico podría enumerar como 5 puntos finales. En consecuencia, si su controlador de host tiene 64 puntos finales disponibles, solo podrá conectar 64/5 dispositivos a ese controlador de host, lo que equivale a 12 dispositivos. Tenga en cuenta que todos los dispositivos USB, incluidos los concentradores USB, requieren terminales, por lo que es importante tener en cuenta esta limitación cuando configure su flujo de trabajo.

Power States

Los estados de energía del dispositivo se denominan D0, D1, D2 y D3. D0 es el estado totalmente encendido y D1, D2 y D3 son estados de bajo consumo. El número de estado está inversamente relacionado con el consumo de energía: los estados con números más altos usan menos energía.

Estado

La API puede devolver las siguientes opciones sobre el estado de la conexión del concentrador.

respuesta de estado	Descripción
inactivo	El concentrador está conectado pero no habla
activo	El concentrador está conectado y en uso
desaparecido	Hub ya no está en el árbol USB
insensible	Hub ya no responde
bloqueado	La conexión del concentrador está bloqueada

11. Inicio sesión

La API puede generar información de registro para todos los eventos USB y almacenar información sobre lo que ha sucedido e información específica del hardware. Esto es útil para ver lo que sucede con la API y puede capturar cualquier falla o problema.

To enable logging you will need to create a config file ending in `.log.cfg`. You can use the following command to create the logging `cfg` file manually:

```
echo*=DEBUG>/etc/opt/cambrionix/cambrionix.log.cfg
```

Luego, después de reproducir el problema, puede comprimir los registros de la carpeta

```
/var/log/cambrionix
```

Puede eliminar el archivo a continuación cuando haya terminado con él.

```
/etc/opt/cambrionix/cambrionix.log.cfg
```

Ubicaciones predeterminadas

Los mensajes de registro generados por `CambrionixApiService` van a `syslog`.

Al usar Windows, los registros se establecerán de manera predeterminada en la siguiente ubicación

```
C:\ProgramData\Cambrionix
```

Al usar macOS, los registros se establecerán de forma predeterminada en la siguiente ubicación

```
Biblioteca>Registros>Cambrionix
```

Al usar Linux, los registros se establecerán de manera predeterminada en la siguiente ubicación

```
/var/log/cambrionix
```

Registro para investigar el comportamiento

Si está experimentando un error o un problema, puede obtener registros del comportamiento para ver con más detalle lo que está sucediendo.

1. Habilitar inicio de sesión en la API
2. Use el concentrador de una manera que cause el problema que está viendo.
3. Espere a que ocurra el problema
4. Tome nota de la hora en que ocurre el problema y luego comprima la carpeta donde se almacenan los registros.

Una vez que tenga esta información, puede revisar los registros usted mismo o, si está en contacto con Cambrionix con respecto al soporte, puede enviar los registros a través del sistema de tickets de soporte.

List of logging options.

There is a list of different options for logs to capture within the Cambrionix Hub API below is a list of all logs that can be enabled directly from the .log.cfg file and what needs to be entered in the file to enable the specific logs.

We would advise to have all logging enabled with the API so if any issue occurs the event is captured in the logs and investigation can take place.

api.battery=DEBUG
api.client=DEBUG
api.client.getset=DEBUG
api.core=DEBUG
api.core.discovery=DEBUG
api.daemon=DEBUG
api.daemon.handle.manager=DEBUG
api.dictionary=DEBUG
api.encoder=DEBUG
api.hub=DEBUG
api.hub.state=DEBUG
api.json=DEBUG
api.json.socket=DEBUG
api.json.websocket=DEBUG
api.serial=DEBUG
api.usbtrees=DEBUG
lib.console=DEBUG
lib.filesystem.watcher=DEBUG
lib.network=DEBUG
lib.service=DEBUG
lib.service.user=DEBUG
lib.settings=DEBUG
lib.task=DEBUG
lib.thread=DEBUG
lib.timer=DEBUG
lib.watchdog=DEBUG

12. muelles

Cuando se han conectado varios productos, con el segundo cargador conectado a un puerto de expansión del primer cargador y así sucesivamente, esto se conoce como Dock. Para algunas operaciones puede ser conveniente tratar estos dos cargadores como una sola unidad que combina los puertos de ambos cargadores.

Para acceder al muelle como una sola unidad, primero llame a `cbrx_discover` con el parámetro `muelles` para obtener la lista de muelles disponibles. Luego llame a `cbrx_connection_open` con la ID en cuestión y también especificando "docks".

La unidad de acoplamiento devolverá el total combinado de claves como `nrOfPorts` y `TotalCurrent_mA`. La gama de puertos se amplía para cubrir el número total combinado de puertos para productos conectados. El producto conectado directamente a la computadora tendrá sus puertos referenciados primero, seguidos por los del cargador conectado al puerto de expansión del primer cargador.

Algunas claves, como `Hardware` o `Firmware`, no se combinan, por lo que estas claves devolverán el valor del cargador principal. Si se desea obtener los valores de estas claves del cargador aguas abajo, es posible abrirlas y recuperarlas de ese cargador de la manera habitual. Abrir un cargador no interfiere con el acceso a la base, excepto cuando se cambia la configuración.

13. Dynamic Hubs

Es posible abrir un concentrador dinámico que sea una combinación de varios otros concentradores. Este se comporta de la misma manera que [muelles](#) hacer. Para abrir un concentrador dinámico, simplemente combine los números de serie de todos los concentradores que desea abrir en un nombre especial con prefijo "Dinámico:" como se muestra en este ejemplo.

```
# Given three Cambrionix hubs with serial numbers of 'AAAAA',
'BBBBB' and 'CCCCC'  handleA =
cbrxapi.cbrx_connection_open("AAAAA") handleB =
cbrxapi.cbrx_connection_open("BBBBB") handleC =
cbrxapi.cbrx_connection_open("CCCCC") handleABC =
cbrxapi.cbrx_connection_open("Dynamic:AAAAA:BBBBB:CCCCC")
print(cbrxapi.cbrx_connection_get(handleA, "nrOfPorts")) # 15
print(cbrxapi.cbrx_connection_get(handleB, "nrOfPorts")) # 8
print(cbrxapi.cbrx_connection_get(handleC, "nrOfPorts")) # 8
print(cbrxapi.cbrx_connection_get(handleABC, "nrOfPorts")) # 31
```

Este concentrador dinámico se trata como una sola entidad con sus puertos numerados del 1 al N, donde N es el número total de puertos en todos los concentradores incluidos.

14. Diccionarios

Para cada concentrador, la API puede devolver dos diccionarios:

- El diccionario Get, que contiene claves que se pueden leer.
- El diccionario Set, que contiene claves que se pueden configurar.

Los pares clave-valor devueltos dependen de los conjuntos de funciones admitidos por la unidad.

14.1. Conjuntos de características

Los siguientes conjuntos de características están disponibles:

Conjunto de características	Descripción
base	Funcionalidad de nivel básico compatible con todas las unidades Cambrionix
sincronizar	Capacidad de sincronización
5 V	La unidad tiene una fuente de alimentación fija de 5V
12 V	La unidad tiene una fuente de alimentación de 12v.
temperatura	La unidad tiene un sensor de temperatura.
PD	La unidad implementa la especificación de suministro de energía USB
puerta	La unidad tiene un producto de control de motor para controlar las puertas de bloqueo para asegurar los dispositivos conectados a los puertos.

Todos los productos admiten el conjunto de características base.

El rango de valores posibles para una clave en el conjunto de funciones básico se puede ampliar si también hay disponible un conjunto de funciones adicional.

La clave de hardware devuelve un valor para el tipo de concentrador.

Estos son los conjuntos de funciones adicionales que admite CambrionixApiService para los distintos tipos de concentrador:

tipo de concentrador devuelto por "Hardware"	sincronizar	5 V	12 V	Temperatura	PD	puerta
PP8C		Sí	Sí	Sí		
PP8S	Sí	Sí	Sí	Sí		
PP15C		Sí	Sí	Sí		
PP15S	Sí	Sí	Sí	Sí		
SS15	Sí	Sí	Sí	Sí		
Serie 8		Sí				
U8C-EXT		Sí	Sí	Sí		
U8C		Sí				
U8RA	Sí	Sí				
U8S-EXT	Sí	Sí	Sí	Sí		
U8S	Sí	Sí				
U10C		Sí				
U10S	Sí	Sí				
U12S	Sí	Sí				
U16S Spade-NL	Sí	Sí				
PD-Sync 4	si*1			Sí	Sí	
ThunderSync2-16	Sí	Sí		Sí		
ThunderSync3-16	Sí	Sí		Sí		
ModIT máx.*2	Sí	Sí		Sí		Sí

*1 Debe tenerse en cuenta que, si bien el PDSync-C4 no implementa el conjunto de funciones de "sincronización" como tal, sí tiene capacidades de sincronización y siempre están disponibles. Esto significa que no hay necesidad de cambiar entre el modo de carga y el modo de sincronización.

*2 El ModIT Max se identificará a sí mismo como ThunderSync3-16, pero tiene hardware adicional para el control de puerta.

14.2. Obtener Diccionario

Llave	Conjunto de características
Adjunto	5 V
Compiled	base
Perfiles habilitados	5 V
Firmware	base
Firmware Requirements	base
FiveVoltRail_flags	5 V
FiveVoltRail_Limit_Max_V	5 V
FiveVoltRail_Limit_Min_V	5 V
FiveVoltRail_V	5 V
FiveVoltRailMax_V	5 V
FiveVoltRailMin_V	5 V
puertas	puerta
Grupo	base
Hardware	base
HardwareBanderas	base
HardwareInformation	base
Salud	base
AnfitriónPresente	PD
InputRail_flags	PD
InputRail_Limit_Max_V	PD
InputRail_Limit_Min_V	PD
InputRail_V	PD
InputRailMax_V	PD

Llave	Conjunto de características
InputRailMin_V	PD
Clave.N	5 V
ModoCambioAuto	sincronizar
nrOfPorts	base
Id. del panel	base
Puerto.N.Batería	sincronizar
Port.N.Current_mA	base
Puerto.N.Descripción	PD
Puerto.N.Energía_Wh	base
Port.N.Banderas	base
Puerto.N.FlashDrive	sincronizar
Port.N.LocationID	sincronizar
Port.N.Fabricante	PD
Puerto.N.Modos	base
Puerto.N.PID	PD
Port.N.ProfileID	5 V
Port.N.Perfiles	5 V
Port.N.SerialNumber	PD
Port.N.TimeCharged_sec	base
Port.N.TimeCharging_sec	base
Puerto.N.USBCadenas	PD
Puerto.N.VID	PD
Puerto.N.Voltaje_10mV	PD
PortInfo.N	base
Información de puertos	base

Llave	Conjunto de características
Perfil.N.habilitado	5 V
pwm_percent	temperatura
reiniciado	base
SeguridadArmado	5 V
Ajustes	base
Título del sistema	base
Temperatura_C	temperatura
Banderas_de_temperatura	temperatura
Temperatura_límite_máx_C	temperatura
TemperaturaMax_C	temperatura
TotalCorriente_mA	5 V
TotalPower_W	5 V
TwelveVoltRail_flags	12 V
TwelveVoltRail_Limit_Max_V	12v
TwelveVoltRail_Limit_Min_V	12v
TwelveVoltRail_V	12 V
TwelveVoltRailMax_V	12 V
TwelveVoltRailMin_V	12 V
Uptime_sec	base

Adjunto

Un campo de bits con un conjunto de bits para cada puerto con un dispositivo conectado, el puerto 1 en el bit 0, el puerto 2 en el bit 1 y así sucesivamente.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Adjunto"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": bit adjunto
}
```

bit adjunto es un valor entero.

Ejemplo, de tres dispositivos conectados a los puertos 1,2,3

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 7
}
```

Compiled

Marca de tiempo de la versión de firmware.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Compilado"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{ "resultado": " fecha de compilación "
}
```

compiled-date is the timestamp of when the Firmware was compiled format "MMM DD YYYY HH mm SS"

marca de tiempo	Descripción
MMM	Mes primeras 3 letras en ingles.
DD	Fecha del mes como un número entero
AAAA	Año como un número entero
S.S	Hora de construcción, 0-23
milímetro	Minuto de construcción, 0-59
SS	Segundo de construcción, 0-59

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": "Jul 08 2015 10:43:20"  
}
```

Perfiles habilitados

Lista de perfiles globales habilitados actualmente

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "Perfiles habilitados"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": " perfiles "  
}
```

perfiles es una lista de todos los perfiles de carga que se aplican al concentrador, los perfiles se muestran como un solo número con un espacio entre cada perfil.

Ejemplo

Valor de retorno donde están habilitados los perfiles de carga 1,2,3 y 4.

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": "1 2 3 4"  
}
```

Firmware

Cadena de versión de firmware.

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "firmware"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": " versión de firmware "  
}
```

firmware-version Es el número de versión del firmware. Formato 'N.nn'

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": "1.55"  
}
```

Firmware Requirements

Obtenga los tipos de firmware aplicables a este concentrador, devueltos como una matriz que contiene información para todos los tipos de firmware que acepta el concentrador. Para cada entrada, el campo de factor de forma indica el tipo de firmware para esa parte, que puede ser "un" para Universal, "pd" para PDSync, "st" para TS3-C10 o "mc" para placa de control de motor.

Firmware	Número de parte	nombre del producto
Universal	PP15S	PowerPad15S
Universal	PP15C	PowerPad15C
Universal	PP8S	PowerPad8S
Universal	SS15	SuperSync15
Universal	TS3-16	ThunderSync3-16
ELEGANTE	TS3-C10	ThunderSync3-C10
Universal	Pala Sub-16	Pala Sub-16
Universal	U8S	U8S
Sincronización PD	PDSync-C4	PDSync-C4
Universal	ModIT-Max	ModIT-Max
Control de motor	Tablero de control de motores	ModIT-Max

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Requisitos de firmware"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

Devuelve una matriz de información sobre cada tipo de firmware actualmente instalado, con detalles de los tipos de firmware necesarios para actualizarlos.

```
{
  "resultado": [
    {
      "Fabricante": " nombre del fabricante ",
      "Hardware": " nombre-producto ",
      "Firmware": " versión de firmware ",
      "Bootloader": " versión del cargador de arranque ",
      "Group": "group-order",
      "FormFactor": " tipo de firmware ",
      "ID de hardware": ID de hardware ,
      "Grupo": " orden de grupo "
      "SerialNumber": "hub-serial",
    }
  ]
}
```

Producción	Descripción
<i>hub-serial</i>	Este es el número de serie del hub devuelto de cbrx_descubrir
<i>Nombre del Fabricante</i>	Nombre definido del fabricante, por defecto es 'Cambrionix'
<i>versión de firmware</i>	Número de versión del firmware. Formato 'N.nn'
<i>versión del gestor de arranque</i>	Número de versión del gestor de arranque. Formato 'N.nn'

Producción	Descripción
<i>orden de grupo</i>	Se utiliza para ordenar concentradores, lo cual es útil cuando se actualizan productos conectados para que los productos posteriores se actualicen y reinicien primero.
<i>tipo de firmware</i>	Se utiliza para indicar qué firmware acepta el producto
<i>Identificación del hardware</i>	número de identificación de hardware del producto del panel frontal
<i>nombre del producto</i>	Nombre de hardware del producto

Ejemplo

```

{
  "jsonrpc": "2.0",
  "id": 0,
  "result": [
    {
      "Manufacturer": "cambrionix",
      "Hardware": "ThunderSync3-16",
      "Firmware": "1.87",
      "Bootloader": "0.21",
      "FormFactor": "un",
      "HardwareID": 50,
      "Group": "-",
      "SerialNumber": "DJ00ASBK"
    },
    {
      "Manufacturer": "cambrionix",
      "Hardware": "Motor Board",
      "Firmware": "0.08",
      "Bootloader": "0.05",
      "FormFactor": "mc",
      "HardwareID": 1,
      "Group": "+",
      "SerialNumber": "DJ00ASBK"
    }
  ]
}

```

FiveVoltRail_flags

Devuelve una lista de indicadores de error del riel de suministro de 5 V que se han detectado, si corresponde.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "FiveVoltRail_flags"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": " bandera "
}
```

Indicador	Descripción
ultravioleta	Se produjo bajo voltaje
VO	Ocurrió sobre voltaje
OV ultravioleta	Ocurrieron tanto bajo como sobrevoltaje.

Ejemplo

```
{
```

```
"jsonrpc": "2.0",  
"id": 0,  
"result": "UV OV"  
}
```

FiveVoltRail_Limit_Max_V

Límite superior del riel de 5V que activará la bandera de error.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "FiveVoltRail_Limit_Max_V"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": FiveVoltRail-Limit-Max
}
```

FiveVoltRail-Limit-Max es un número decimal en voltios, formato n.nn

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 5.58
}
```

FiveVoltRail_Limit_Min_V

Límite inferior del riel de 5V que activará la bandera de error

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "FiveVoltRail_Limit_Min_V"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": FiveVoltRail-Limit-Min
}
```

FiveVoltRail-Limit-Min es un número decimal en voltios, formato n.nn

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 3.50
}
```

FiveVoltRail_V

Tensión de alimentación actual de 5 V en voltios (V)

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "FiveVoltRail_V"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": FiveVoltRail_V  
}
```

FiveVoltRail_V es un número decimal en voltios, formato n.nn

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": 5.25  
}
```

FiveVoltRailMax_V

Tensión de alimentación de 5 V más alta medida en voltios (V)

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "FiveVoltRailMax_V"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": FiveVoltRailMax_V
}
```

FiveVoltRailMax_V es un número decimal en voltios, formato n.nn

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 5.25
}
```

FiveVoltRailMin_V

Tensión de alimentación de 5 V más baja medida en voltios (V)

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "FiveVoltRailMin_V"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": FiveVoltRailMin_V  
}
```

FiveVoltRailMin_V es un número decimal en voltios, formato n.nn

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": 5.20  
}
```

puertas

Devuelve un objeto que describe los estados de todas las puertas en el producto de expansión, si está presente. Actualmente, esto solo está disponible en la gama ModIT.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Puertas"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado":
  {
    " N ":" estado "
  }
}
```

N es el número de puerto

state	Descripción
abierto	la puerta esta abierta
cerrado	la puerta esta cerrada
apertura	la puerta se abre

state	Descripción
clausura	la puerta se esta cerrando
estancado	La puerta ha dejado de moverse.
se acabó el tiempo	La puerta ha tardado demasiado en responder
desconocido	Ni los interruptores de extremo superior ni inferior están activados
establo abierto	La puerta se ha estancado en la posición abierta
cerrado	La puerta se ha estancado en la posición cerrada
apertura estancada	La puerta se ha atascado mientras se abría
cierre estancado	La puerta se ha estancado mientras se cerraba
desactivado	Se ha establecido un indicador de deshabilitado que impide que la puerta se abra/cierre

Ejemplo

```

{
  "jsonrpc": "2.0",
  "id": 0,
  "result": {
    "1": "open",
    "2": "closed",
    "3": "closed",
    "4": "opening"
  }
}

```

Grupo

Carta de grupo leída de los puentes de PCB.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Grupo"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": " orden de grupo "
}
```

group-order Se utiliza para ordenar concentradores, lo cual es útil cuando se actualizan productos conectados para que los productos posteriores se actualicen y reinicien primero., o "-" si no se instaló un puente de grupo.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": "-"
}
```

Hardware

Número de pieza del concentrador.

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "Hardware"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": " nombre-producto "  
}
```

product-name es el nombre de hardware del producto

Ejemplo

```
{  
  "result": "ThunderSync3-16"  
}
```

HardwareBanderas

Banderas que indican si las características están presentes

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Banderas de hardware"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": " bandera "
}
```

Indicador	Descripción
S	Conjunto de funciones de sincronización
L	Conjunto de características de 5V
mi	conjunto de funciones de 12 V
T	Conjunto de funciones de temperatura
P	Conjunto de características de PD

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": "SLET"  
}
```

HardwareInformation

Information on the hub

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "HardwareInformation"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": {
    "ProductName": "product-name",
    "ProductWebPage": "product-webpage",
    "TemperatureRangeC": {
      "Min": min-temperature,
      "Max": max-temperature
    },
    "HumidityRange": {
      "Min": min-humidity,
      "Max": max-humidity
    },
    "DimensionsMillimetres": {
```

```

"Width": product-width,
"Length": product-length,
"Height": product-height
    },
"HostPortType": "host-port",
"HostPortBandwidth": "host-port-bandwidth",
"HubMaxPowerOutputWatts": hub-max-power-output,
"Ports": {
  "1": {
    "HardwareInformation": {
      "Type": "port-type",
      "Bandwidth": "port-bandwidth",
      "VoltageMax": port-max-volts,
      "MilliampsMax": port-max-current
    }
  },
  // etc all ports.
}
}
}
}

```

Variable	Descripción
<i>nombre del producto</i>	Product name of the hub
<i>product-webpage</i>	The webpage for the hub
<i>min-temperature</i>	The minimum recommended temperature for the hubs environment (°C)

Variable	Descripción
<i>max-temperature</i>	The maximum recommended temperature for the hubs enviroment
<i>min-humidity</i>	The minimum ambient humidty % for the hubs enviroment
<i>max-humidity</i>	The maximum ambient humidity % for the hubs enviroment
<i>product-width</i>	The width of the hub (mm)
<i>product-length</i>	The length of the hub (mm)
<i>product-height</i>	The height of the hub (mm)
<i>host-port</i>	The USB type of the host port
<i>host-port-bandwidth</i>	The maximum bandwidth for the host port (Gbps)
<i>hub-max-power-output</i>	The maximum power output for the whole hub
<i>port-type</i>	The USB type of the downstream ports
<i>port-bandwidth</i>	The maximum bandwidth for the downstream ports (Gbps)
<i>port-max-volts</i>	The maximum Voltage output for the downstream ports (V)
<i>port-max-current</i>	The maximum Current output fo the downstream ports (mA)

Ejemplo

```

{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "ProductName": "ThunderSync3-C10",
    "ProductWebPage": "https://www.cambrionix.com/products/thundersync3-c10",
    "TemperatureRangeC": {
      "Min": 10,
      "Max": 35
    },
  },
  "HumidityRange": {
    "Min": 5,
    "Max": 95
  }
}

```

```
    },  
    "DimensionsMillimetres": {  
      "Width": 193,  
      "Length": 136,  
      "Height": 34  
    },  
    "HostPortType": "Thunderbolt 3",  
    "HostPortBandwidth": "40Gbps",  
    "HubMaxPowerOutputWatts": 150,  
    "Ports": {  
      "1": {  
        "HardwareInformation": {  
          "Type": "USB Type-C",  
          "Bandwidth": "5Gbps",  
          "VoltageMax": 5.2,  
          "MilliampsMax": 3000  
        }  
      },  
      // etc all ports.  
    }  
  }  
}
```

Salud

Todas las claves disponibles que no son específicas del puerto y cambian dinámicamente, como un diccionario.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Salud"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": {
    "Tiempo de actividad_segundo": Tiempo de actividad_segundo,
    "FiveVoltRail_V": FiveVoltRail_V ,
    "FiveVoltRailMin_V": FiveVoltRailMin_V ,
    "FiveVoltRailMax_V": FiveVoltRailMax_V ,
    "FiveVoltRail_flags": " FiveVoltRail_flags ",
    "TwelveVoltRail_V": DoceVoltRail_V ,
    "TwelveVoltRailMin_V": DoceVoltRailMin_V ,
    "TwelveVoltRailMax_V": DoceVoltRailMax_V ,
    "InputRail_V": InputRail_V
    "InputRailMin_V": InputRailMin_V ,
```

```

"InputRailMax_V": InputRailMax_V ,
"TwelveVoltRail_flags": " TwelveVoltRail_flags ",
"InputRail_flags": " InputRail_flags ",
"Temperatura_C": Temperatura_C ,
"TemperaturaMax_C": TemperatureMax_C ,
"Temperature_flags": " Temperature_flags ",
"Reiniciado": Reiniciado
}
}

```

Producción	Descripción
<i>Uptime_sec</i>	ver Uptime_sec
<i>FiveVoltRail_V</i>	ver FiveVoltRail_V
<i>FiveVoltRailMin_V</i>	ver FiveVoltRailMin_V
<i>FiveVoltRailMax_V</i>	ver FiveVoltRailMax_V
<i>FiveVoltRail_flags</i>	ver FiveVoltRail_flags
<i>TwelveVoltRail_V</i>	ver TwelveVoltRail_V
<i>TwelveVoltRailMin_V</i>	ver TwelveVoltRailMin_V
<i>TwelveVoltRailMax_V</i>	ver TwelveVoltRailMax_V
<i>InputRail_V</i>	ver InputRail_V
<i>InputRailMin_V</i>	ver InputRail_Limit_Min_V
<i>InputRailMax_V</i>	ver InputRailMax_V
<i>TwelveVoltRail_flags</i>	ver TwelveVoltRail_flags
<i>InputRail_flags</i>	ver InputRail_flags
<i>Temperatura_C</i>	ver Temperatura_C
<i>TemperaturaMax_C</i>	ver TemperaturaMax_C

Producción	Descripción
<i>Banderas_de_temperatura</i>	ver Banderas_de_temperatura
<i>reiniciado</i>	ver reiniciado

Ejemplo

```

{
  "jsonrpc": "2.0",
  "id": 0,
  "result": {
    "Uptime_sec": 528422,
    "FiveVoltRail_V": 5.23,
    "FiveVoltRailMin_V": 5.14,
    "FiveVoltRailMax_V": 5.25,
    "FiveVoltRail_flags": "",
    "TwelveVoltRail_V": 12.12,
    "TwelveVoltRailMin_V": 11.99,
    "TwelveVoltRailMax_V": 12.2,
    "InputRail_V": 12.12,
    "InputRailMin_V": 11.99,
    "InputRailMax_V": 12.2,
    "TwelveVoltRail_flags": "",
    "InputRail_flags": "",
    "Temperature_C": 37.3,
    "TemperatureMax_C": 41.1,
    "Temperature_flags": "",
    "Rebooted": true
  }
}

```

AnfitriónPresente

El concentrador monitorea la toma USB del host en busca de una computadora host conectada.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Anfitrión presente"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": anfitrión
}
```

host	descripción
verdadero	Se detecta el host
FALSO	No se detecta ningún host

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": true
}
```

InputRail_flags

Lista de indicadores de error del riel de entrada, si se establece alguno.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "InputRail_flags"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": bandera
}
```

Indicador	Descripción
ultravioleta	se produjo bajo voltaje
VO	se produjo sobretensión
sin banderas	el voltaje es aceptable

Ejemplo

```
{
  "jsonrpc": "2.0",
```

```
"id": 0,  
"result": "OV UV"  
}
```

InputRail_Limit_Max_V

Límite superior del riel de entrada que activará el indicador de error

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "InputRail_Limit_Max_V"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": InputRail-Limit-Max  
}
```

InputRail-Limit-Max es un número decimal en voltios, formato n.nn

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": 24.7  
}
```

InputRail_Limit_Min_V

Límite inferior del riel de entrada que activará el indicador de error.

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "InputRail_Limit_Min_V"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": InputRail-Limit-Min  
}
```

InputRail-Limit-Min es un número decimal en voltios, formato n.nn

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": 9.59  
}
```

InputRail_V

Suministro de riel de entrada de corriente en voltios (V).

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "InputRail_V"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": InputRail_V
}
```

InputRail_V es un número decimal en voltios, formato n.nn

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 24.03
}
```

InputRailMax_V

Voltaje de entrada más alto medido en voltios (V).

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "InputRailMax_V"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": InputRailMax_V
}
```

InputRailMax_V es un número decimal en voltios, formato n.nn

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 24.14
}
```

InputRailMin_V

Voltaje de entrada más bajo medido en voltios (V).

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "InputRailMin_V"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": InputRailMin_V
}
```

es un número decimal en voltios, formato n.nn

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 23.82
}
```

Clave.N

Obtenga información si se ha presionado un botón, los doble clic no se pueden detectar.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Clave.N"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": botón
}
```

botón	Descripción
0	el botón n no ha sido presionado desde la última vez que se leyó esta entrada
1	el botón n ha sido presionado desde la última vez que se leyó esta entrada

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 0
}
```

ModoCambioAuto

El cambio de modo de carga a sincronización es automático.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "ModoCambioAuto"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": cambio automático
}
```

cambio automático	Descripción
verdadero	El cambio de modo de carga a sincronización es automático.
FALSO	El cambio de modo de carga a sincronización es manual.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": true
}
```

nrOfPorts

Número de puertos USB en el concentrador.

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "nrOfPorts"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": número de puertos  
}
```

portsnumber es un número entero de la cantidad de puertos disponibles

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": 8  
}
```

Nota

- En el PDSync-C4 tendrá un puerto adicional 0, que es la información del puerto host.

Id. del panel

Número de PanelID de la placa del panel frontal, si está instalado.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "ID del panel"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": " ID de panel "
}
```

Panel-ID es el número de ID del producto del panel frontal; si no está instalado, devolverá Ausente/Ninguno

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": "Absent"
}
```

Puerto.N.Batería

Si es posible, recupere el nivel actual de la batería del dispositivo conectado. Consulte las notas sobre la recopilación de información de la batería. Según el tipo de dispositivo (Android™, iOS, etc.) y el sistema operativo anfitrión, es posible que se devuelvan datos diferentes.

Para Apple, se debe instalar "Apple Mobile Device Support" (incluido con iTunes)

Para Android, adb debe estar instalado y en ejecución.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    empuñadura de conexión
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": {
    "CurrentLevel": nivel de corriente de la batería ,
    "HoraActual": hora-del-centro-actual ,
    "StartingLevel": nivel de inicio de carga ,
    "StartingTime": tiempo de inicio de carga ,
  }
}
```

Producción	Descripción
<i>nivel de corriente de la batería</i>	El nivel actual de la batería del dispositivo se muestra como un porcentaje
<i>tiempo-del-centro-actual</i>	El tiempo del concentrador, que se muestra como un número entero en ms
<i>nivel de inicio de carga</i>	Nivel de porcentaje de batería cuando el dispositivo está conectado
<i>tiempo de inicio de carga</i>	Se inició la carga del tiempo del concentrador, que se muestra como un número entero en ms

Ejemplo

```

{
  "jsonrpc": "2.0",
  "id": 0,
  "result": {
    "CurrentLevel": 78,
    "CurrentTime": 15234254346,
    "StartLevel": 23,
    "StartTime": 15124151512,
  }
}

```

Port.N.Current_mA

Corriente que se envía al dispositivo USB conectado a este puerto USB en miliamperios (mA).

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Port.N.Current_mA"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": Current_mA
}
```

Current_mA es la corriente que se entrega al dispositivo, en mA (miliamperios)

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 0
}
```

Puerto.N.Descripción

Descripción según lo informado por el dispositivo USB conectado a este puerto USB si se pudo detectar. Se devuelve una cadena vacía si no se pudo detectar.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Puerto.N.Descripción"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": " Descripción "
}
```

Descripción es el nombre del hardware.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": "SuperPhone6"
}
```

Puerto.N.Energía_Wh

Se ha consumido energía en el dispositivo USB de este puerto USB.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Port.N.Energy_Wh"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": energía-Wh
}
```

energy-Wh es la potencia en vatios-hora (calculada cada segundo), formato de nn

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 0.0
}
```

Port.N.Banderas

Obtenga una lista de todas las banderas en un puerto específico

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Port.N.Flags"
  ]
}
```

manga de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": " bandera "
}
```

Lista de caracteres de indicador sensibles a mayúsculas y minúsculas, separados por espacios. O, S, B, I, P, C, F son mutuamente

exclusivo. A y D son mutuamente excluyentes.

Indicador	Descripción
O	El puerto está en modo DESACTIVADO
S	El puerto está en modo de SINCRONIZACIÓN
B	El puerto está en modo sesgado
I	El puerto está en modo de carga y está INACTIVO

P	El puerto está en modo de carga y está PERFILANDO
C	El puerto está en modo de carga y está CARGANDO
F	El puerto está en modo de carga y ha TERMINADO de cargar
A	El dispositivo está CONECTADO a este puerto
D	No hay ningún dispositivo conectado a este puerto. El puerto está DESCONECTADO
T	El dispositivo ha sido robado del puerto: ROBO
mi	Se detectan ERRORES. Véase el comando health
R	El sistema se ha REINICIALIZADO. Véase el comando crf
r	El Vbus se está reiniciando durante el cambio de modo

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": "R D S"
}
```

Puerto.N.FlashDrive

Si se detecta, devuelve el punto de montaje de una unidad flash USB. Para Windows, será una letra de unidad; de lo contrario, será un punto de montaje de volumen.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Port.N.FlashDrive"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "Ruta:": " ubicación ",
  "Capacidad": memoria total ,
  "Disponible": memoria disponible
}
```

Producción	Descripción
<i>ubicación</i>	Ubicación del disco en el sistema host
<i>memoria total</i>	Memoria total en disco
<i>memoria disponible</i>	No se utiliza la memoria en el disco

Ejemplos:

Ventanas:

```
{
  "json": "2.0",
  "id": 0,
  "Path": "H:",
  "Capacity": 123123123,
  "Available": 123123
}
```

macOS®:

```
{
  "json": "2.0",
  "id": 0,
  "Path": "/Volumes/SanDisk1",
  "Capacity": 123123123,
  "Available": 123123
}
```

Linux:

```
{
  "json": "2.0",
  "id": 0,
  "Path": "/media/bob/SanDisk1",
  "Capacity": 123123123,
  "Available": 123123
}
```

Si no hay una unidad flash, el valor devuelto simplemente será falso.

Esta misma información también se proporcionará en PortsInfo, PortInfo.N o cbrx_discover('all') en un campo "FlashDrive" donde corresponda y esté presente. Si no es aplicable o está preestablecido, este campo estará ausente.

Port.N.LocationID

Devuelve el ID de ubicación de un puerto específico. Esto no requiere que se conecte un dispositivo y, por lo tanto, se puede usar para identificar de manera única una ranura USB. Los ID de ubicación indican el número de bus en el que se encuentra un controlador de host USB en el primer byte, luego los números de puerto hacia abajo en el árbol para dispositivos secundarios.

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "Port.N.LocationID"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": ubicación-id  
}
```

id-ubicación es el [Location IDs](#) como un entero

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": 563154944  
}
```

Nota

- for USB3 hubs, this location ID will be different when a USB3 device is plugged in compared to a USB2 device

Port.N.Fabricante

Fabricante según lo informado por el dispositivo USB conectado a este puerto USB, si pudiera detectarse. Se devuelve una cadena vacía si no se pudo detectar.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Port.N.Fabricante"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": " fabricante del dispositivo "
}
```

fabricante del dispositivo es el nombre del fabricante del dispositivo

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": "SuperPhone Makers Inc."
}
```

Puerto.N.Modo

Modo de puerto actual.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    empuñadura de conexión
    "Port.N.Mode"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": " modo "
}
```

Para concentradores USB estándar, el modo puede ser cualquiera de los siguientes:

carácter de modo	Descripción
s	Modo de sincronización
c	modo de carga
b	modo sesgado
o	Apagado

Para concentradores tipo C, el modo puede ser:

carácter de modo	Descripción
c	Los sistemas compatibles cuentan con una interfaz para "forzar" la alimentación
o	Apagado

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": "s"  
}
```

Puerto.N.PID

ID de producto del dispositivo USB conectado a este puerto USB, si se pudo detectar.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Puerto.N.PID"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": identificación del producto
}
```

product-id es el número de identificación del producto o PID. Se muestra como un número entero. Se devuelve 0 (cero) si no se pudo detectar.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 0
}
```

Port.N.ProfileID

Número de identificación del perfil.

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "Puerto.N.PID"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": perfil-ID  
}
```

ID de perfil es el número de perfil, o 0 si no se está cargando.

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": 0  
}
```

Port.N.Perfiles

Lista de perfiles habilitados para este puerto.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Port.N.Perfiles"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": " perfiles "
}
```

perfiles es una lista de todos los perfiles de carga que se aplican al puerto, los perfiles se muestran como un solo número con un espacio entre cada perfil.

Ejemplo

los perfiles de carga 1,2,3 y 4 están habilitados.

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": "1 2 3 4"
}
```

Port.N.SerialNumber

Número de serie según lo informado por el dispositivo USB conectado a este puerto USB, si pudiera detectarse. Se devuelve una cadena vacía si no se pudo detectar.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Port.N.SerialNumber"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "result": "usb-serial"
}
```

usb-serial is the USB serial number

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": "1127dfa9037s1a8cb1"
}
```

Port.N.TimeCharged_sec

Tiempo en segundos desde que este puerto USB detectó que el dispositivo completó la carga.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Port.N.TimeCharged_sec"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": carga completa
}
```

charge-complete es el tiempo en segundos que ha pasado desde que el dispositivo completó la carga, se devolverá -1 si este puerto no ha detectado una carga completa.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 0
}
```

Port.N.TimeCharging_sec

Tiempo en segundos desde que este puerto USB comenzó a cargar un dispositivo conectado. Se devolverá 0 si el puerto USB no ha comenzado a cargar un dispositivo conectado.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Port.N.TimeCharging_sec"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": tiempo de carga
}
```

el tiempo de carga es el tiempo transcurrido en segundos que un puerto ha estado cargando un dispositivo.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 0
}
```

Puerto.N.USB Cadenas

Un diccionario que contiene los valores de "Fabricante", "Descripción" y "Número de serie" para este puerto USB.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Puerto.N.Cadenas USB"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "SerialNumber": "usb-serial",
  "Descripción": " descripción ",
  "Fabricante": " fabricante del dispositivo "
}
```

Producción	Descripción
<i>usb-serial</i>	USB serial number
<i>descripción</i>	Nombre del equipo
<i>fabricante del dispositivo</i>	El nombre del fabricante del dispositivo.

Ejemplo

```
{  
  "json": 2.0,  
  "id": 0,  
  "SerialNumber": "23213dfe12e2412",  
  "Description": "SuperPhone6",  
  "Manufacturer": "SuperPhone Makers Inc."  
}
```

Puerto.N.VID

Id. de proveedor del dispositivo USB conectado a este puerto USB.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Puerto.N.VID"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": Id. de proveedor
}
```

Vendor-ID is the vendor ID, VID. Se muestra como un número entero. Se devuelve 0 (cero) si no se pudo detectar.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 0
}
```

Puerto.N.Voltaje_10mV

Voltaje que se suministra al puerto en 10 mV.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Puerto.N.Voltaje_10mV"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": voltaje-10mV
}
```

voltage-10mV es el voltaje suministrado al puerto como un número entero en incrementos de 10mV.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 520
}
```

PortInfo.N

Obtenga toda la información del puerto para el puerto especificado. Todas las claves y valores disponibles para este puerto como diccionario.

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "PortInfo.N"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado":{  
    "Puerto": número de puerto ,  
    "Corriente_mA": Corriente_mA ,  
    "LocationID": ubicación-id ,  
    "Banderas": banderas ,  
    "USBVersion": versión usb ,  
    "VID": Id. de proveedor ,  
    "PID": ID de producto ,  
    "Fabricante": " fabricante del dispositivo ",  
    "Descripción": " descripción ",  
    "SerialNumber": "usb-serial",
```

```

"Árbol USB": {
  "LocationID": ubicación-id ,
  "USBVersion": versión usb ,
  "Alimentación USB": {
    "Estado": " estado de poder ",
    "Descripción": " descripción de poder "
      },
  "Velocidad USB": {
    "Velocidad": " velocidad USB ",
    "Descripción": "Descripción USB "
    "Capacidad": {
      "Velocidad": " velocidad capaz ",
      "Descripción": " descripción capaz "
        }
      },
  "Puntos finales": {
    "Activo": puntos finales activos ,
    "Memoria": memoria de punto final
      }
    }
  }
}

```

Producción	Descripción
<i>número de puerto</i>	El número del puerto en el concentrador
<i>Actual_mA</i>	Corriente que se entrega al dispositivo, en mA (miliamperios)
<i>id-ubicación</i>	El Location IDs como un entero

Producción	Descripción
<i>banderas</i>	Banderas en el puerto, ver Port.N.Banderas
<i>versión usb</i>	El número de versión USB de la conexión al concentrador. Formato 'N.nn'
<i>Id. de proveedor</i>	Número de ID de vedor del dispositivo o VID. Mostrado como un entero
<i>ID del Producto</i>	Número de identificación del producto o PID. Mostrado como un entero
<i>fabricante del dispositivo</i>	El nombre del fabricante del dispositivo.
<i>descripción</i>	Nombre del equipo
<i>usb-serial</i>	USB serial number
<i>estado de poder</i>	USB Power States código
<i>poder-descripción</i>	USB power turned on/off
<i>Velocidad USB</i>	Maximum speed USB connection capable of
<i>USB-descripción</i>	Nombre de la conexión USB, es decir USB de supervelocidad de 5 Gbps
<i>velocidad capaz</i>	Dispositivo de máxima velocidad de datos capaz de
<i>descripción capaz</i>	Nombre del dispositivo de máxima velocidad de datos capaz de
<i>puntos finales activos</i>	Cuántos puntos finales está usando el dispositivo
<i>memoria de punto final</i>	Cantidad de memoria utilizada por los puntos finales

Ejemplo

```
{
```

```
"jsonrpc": "2.0",
"id": 0,
"result": {
  "Port": 1,
  "Current_mA": 1084,
  "LocationID": 563154944,
  "Flags": "R A S",
  "USBVersion": 2.1,
  "VID": 1256,
  "PID": 26720,
  "Manufacturer": "SAMSUNG",
  "Description": "SAMSUNG_Android",
  "SerialNumber": "RFCN20Q8LJM",
  "USBTree": {
    "LocationID": 563154944,
    "USBVersion": 2.1,
    "USBPower": {
      "State": "D0",
      "Description": "On"
    },
    "USBSpeed": {
      "Speed": "480Mbps",
      "Description": "High",
      "Capability": {
        "Speed": "10Gbps",
        "Description": "SuperSpeed USB 10Gbps"
      }
    },
    "Endpoints": {
      "Active": 9,
      "Memory": 36864
    }
  }
}
```

Información de puertos

Obtenga toda la información del puerto para todos los puertos. Toda la información disponible para todos los puertos como diccionario de diccionarios. La mayoría de estos valores se pueden consultar individualmente

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "Información de puertos"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones

```
{  
  "Puerto.1": {  
    "Puerto": número de puerto ,  
    "Corriente_mA": Corriente_mA ,  
    "Banderas": banderas ,  
    "ID de perfil": ID de perfil ,  
    "TimeCharging_sec": tiempo de carga ,  
    "TimeCharged_sec": carga completa ,  
    "Energy_Wh": energía-Wh ,  
    "VID": Id. de proveedor ,  
    "PID": ID de producto ,
```

```

"Fabricante": " fabricante del dispositivo ",
"Descripción": " descripción ",
"SerialNumber": "usb-serial",
"PhoneSerialNumber": " teléfono-serie ",
"PhoneIdentity": " nombre-teléfono ",
"IMEI": " Número IMEI ",
"MacAddress": " MacAddress ",
"PhoneSoftwareVersion": " versión del sistema operativo del
teléfono "
"Árbol USB": {
"USB2": {
"LocationID": ubicación-id ,
"VID": Id. de proveedor ,
"PID": ID de producto ,
"Fabricante": " fabricante del dispositivo ",
"Descripción": " descripción ",
"SerialNumber": "usb-serial",
"USBVersion": versión usb ,
"Batería": {
"DataSource": " batería-fuente-de-datos ",
"TrustLevel": " nivel de confianza ",
"PairingSupported": emparejamiento compatible ,
"HealthPercent": estado de la batería ,
"CurrentLevel": nivel de corriente de la batería ,
"HoraActual": hora-del-centro-actual ,
"StartingLevel": nivel de inicio de carga ,
"StartingTime": tiempo de inicio de carga ,
"CapacityNew": nueva-batería-capacidad ,

```

```

"Capacidad": capacidad actual de la batería ,
"ChargingStatus": " estado de carga ",
    },
"PhoneSerialNumber": " teléfono-serie ",
"PhoneIdentity": " nombre-teléfono ",
"IMEI": " Número IMEI ",
"MacAddress": " MacAddress ",
"PhoneSoftwareVersion": " versión del sistema operativo del
teléfono "
    }
},
"Batería": {
"DataSource": " batería-fuente-de-datos ",
"TrustLevel": " nivel de confianza ",
"PairingSupported": emparejamiento compatible ,
"CurrentLevel": nivel de corriente de la batería ,
"HoraActual": hora-del-centro-actual ,
"StartingLevel": nivel de inicio de carga ,
"StartingTime": tiempo de inicio de carga ,
"CapacityNew": nueva-batería-capacidad ,
"Capacidad": capacidad actual de la batería ,
"ChargingStatus": " estado de carga ",
"HealthPercent": estado de la batería
    }
}
}

```

Producción	Descripción
<i>número de puerto</i>	El número del puerto en el concentrador
<i>Actual_mA</i>	Corriente que se suministra al dispositivo móvil, en mA (miliamperios)
<i>banderas</i>	Banderas en el puerto, ver Port.N.Banderas
<i>Perfil Id</i>	El número de perfil, o 0 si no está cargando.
<i>tiempo de carga</i>	Tiempo transcurrido en segundos que un puerto ha estado cargando un dispositivo.
<i>carga completa</i>	Tiempo en segundos desde que el dispositivo completó la carga
<i>energía-Wh</i>	Potencia en vatios-hora (calculada cada segundo), formato de nn
<i>Id. de proveedor</i>	Número de ID de vedor del dispositivo o VID. Mostrado como un entero
<i>ID del Producto</i>	Número de identificación del producto o PID. Mostrado como un entero
<i>fabricante del dispositivo</i>	El nombre del fabricante del dispositivo.
<i>descripción</i>	Nombre del equipo
<i>usb-serial</i>	USB serial number
<i>teléfono-serie</i>	Número de serie del teléfono
<i>nombre de teléfono</i>	Nombre del teléfono
<i>número IMEI</i>	El número IMEI del teléfono.
<i>Dirección MAC</i>	Una dirección única asignada al dispositivo móvil. Es un valor de 48 bits, que consta de doce caracteres hexadecimales
<i>versión del sistema operativo del teléfono</i>	Número de versión del sistema operativo en el teléfono

Producción	Descripción
<i>id-ubicación</i>	El Location IDs como un entero
<i>versión usb</i>	El número de versión USB de la conexión al concentrador. Formato 'N.nn'
<i>batería-fuente-de-datos</i>	La fuente de la información de la batería del dispositivo
<i>nivel de confianza</i>	Si el dispositivo es de confianza/emparejado
<i>emparejamiento de apoyo</i>	Lo que el dispositivo admita para ser confiable/emparejado
<i>salud de la batería</i>	El estado de la batería se muestra como un porcentaje
<i>nivel de corriente de la batería</i>	El nivel actual de la batería del dispositivo se muestra como un porcentaje
<i>tiempo-del-centro-actual</i>	El tiempo del concentrador, que se muestra como un número entero en ms
<i>nivel de inicio de carga</i>	Nivel de porcentaje de batería cuando el dispositivo está conectado
<i>tiempo de inicio de carga</i>	Se inició la carga del tiempo del concentrador, que se muestra como un número entero en ms
<i>nueva-bateria-capacidad</i>	La capacidad de la batería del dispositivo de nuevo
<i>capacidad actual de la batería</i>	La capacidad de la batería del dispositivo ahora
<i>estado de carga</i>	Estado de carga de la batería, es decir lleno
<i>emparejamiento de apoyo</i>	Lo que el dispositivo admita para ser confiable/emparejado
<i>salud de la batería</i>	El estado de la batería se muestra como un porcentaje

Ejemplo

Ejemplo recortado de información devuelta.

```

{
  "json": "2.0",
  "id": 0,
  "Port.1": {
    "Port": 1,
    "Current_mA": 126,
    "Flags": "R A S",
    "ProfileID": 0,
    "TimeCharging_sec": 0,
    "TimeCharged_sec": 0,
    "Energy_Wh": 0.0,
    "VID": 1452,
    "PID": 4776,
    "Manufacturer": "SuperPhone Makers Inc.",
    "Description": "SuperPhone",
    "SerialNumber": "012a37d1fa07617ad7ef0430ba49f479ab9fb6b8",
    "PhoneSerialNumber": "ZCZCZCZCZCZC",
    "PhoneIdentity": "My Old SuperPhone",
    "IMEI": "354430099009999",
    "MacAddress": "aa:bb:cc:ff:ee:ff",
    "PhoneSoftwareVersion": "12.4.8",
    "USBTree": {
      "USB2": {
        "LocationID": 589570048,
        "VID": 1452,
        "PID": 4776,
        "Manufacturer": "SuperPhone Makers Inc.",
        "Description": "SuperPhone",
        "SerialNumber": "012a37d1fa07617ad7ef0430ba49f479ab9fb6b8",
        "USBVersion": 2.0,
        "Battery": {
          "DataSource": "imobiledevice",
          "TrustLevel": "paired",
          "PairingSupported": true,
          "HealthPercent": 95,
          "CurrentLevel": 100,
          "CurrentTime": 1613056296,
          "StartingLevel": 100,
          "StartingTime": 1613056293,
          "CapacityNew": 1751,
          "Capacity": 1678,
          "ChargingStatus": "full"
        },
        "PhoneSerialNumber": "ZCZCZCZCZCZC",
        "PhoneIdentity": "My Old SuperPhone",
        "IMEI": "354430099009999",
        "MacAddress": "aa:bb:cc:ff:ee:ff",
        "PhoneSoftwareVersion": "12.4.8"
      }
    }
  },
  "Battery": {
    "DataSource": "imobiledevice",
    "TrustLevel": "paired",
    "PairingSupported": true,
    "HealthPercent": 95,
  }
}

```

```
        "CurrentLevel": 100,  
        "CurrentTime": 1613056296,  
        "StartingLevel": 100,  
        "StartingTime": 1613056293,  
        "CapacityNew": 1751,  
        "Capacity": 1678,  
        "ChargingStatus": "full"  
    }  
},  
"Port.2": {  
    "Port": 2,  
    "Current_mA": 0,  
    "Flags": "R D S",  
    "ProfileID": 0,  
    "TimeCharging_sec": 0,  
    "TimeCharged_sec": 0,  
    "Energy_Wh": 0.0,  
    "VID": 0,  
    "PID": 0,  
    "Manufacturer": "",  
    "Description": "",  
    "SerialNumber": ""  
},  
"Port.3": ...  
}
```

Perfil.N.habilitado

Obtenga información si un perfil específico está habilitado. Consulte los manuales de usuario de productos específicos sobre los perfiles disponibles en su concentrador.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Perfil.N.habilitado"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": perfil habilitado
}
```

perfil habilitado	Descripción
verdadero	El perfil específico está habilitado
FALSO	El perfil específico no está habilitado

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": false
}
```

pwm_percent

Velocidad del ventilador.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "pwm_percent"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": porcentaje de fans
}
```

fan-percent es la velocidad del ventilador como un porcentaje, que se muestra como un número 0-100

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 100
}
```

reiniciado

Una bandera que indica si el sistema se ha reiniciado desde que se encendió.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Reiniciado"
  ]
}
```

manga de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": reiniciado
}
```

reiniciado	Descripción
verdadero	el sistema ha sido reiniciado
FALSO	no se ha producido ningún reinicio.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": true
}
```

SeguridadArmado

¿La seguridad está armada?

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Seguridad Armada"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": seguridad armada
}
```

armado de seguridad	Descripción
verdadero	La seguridad ha sido armada
FALSO	La seguridad no ha sido armada.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": false
}
```

Ajustes

Obtain current hub Internal hub settings.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Ajustes",
    verdadero
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": {
    "company_name": " nombre del fabricante ",
    "nombre_producto": " nombre-producto ",
    "nombre_local": " nombre-local ",
    "attach_threshold": " adjuntar-umbral ",
    "perfil_predeterminado": [ perfil-predeterminado ],
    "remap_ports": [ orden de puerto ],
    "ports_on": [ ports-on ],
    "sync_chrg": [ carga sincronizada ],
    "alt_sync_chrg": [ alt-sync-carga ],
    "misc_flags": Internal hub-flags,
  }
}
```

```

"display_mode": " modo de visualización ",
"charged_threshold": " charged-threshold ",
"temperature_max": " temperatura de apagado ",
"escalonar": " escalonar "
  }
}

```

Variables	Descripción
<i>Nombre del Fabricante</i>	Nombre definido del fabricante, por defecto es 'Cambrionix'
<i>nombre del producto</i>	Nombre de hardware del producto
<i>nombre local</i>	Local name set by the user, "-" if not set
<i>adjuntar-umbral</i>	Consumo de corriente en mA que el concentrador detecta que un dispositivo está conectado, "d" significa que el valor predeterminado de fábrica está configurado
<i>perfil por defecto</i>	Perfil predeterminado para cada puerto, lista separada por comas
<i>orden de puerto</i>	Ordene los puertos por número de puerto, lista separada por comas
<i>puertos activados</i>	Si cada puerto está activado de forma predeterminada, 0 está desactivado de forma predeterminada 1 está activado de forma predeterminada, lista separada por comas
<i>sincronización de carga</i>	Si CDP* en cada puerto está activado, 0 está desactivado, 1 está activado, lista separada por comas
<i>alt-sync-carga</i>	Si CDP* alternativo en cada puerto activado, 0 está desactivado, 1 está activado, lista separada por comas
<i>Internal hub-flags</i>	If any Internal hub Misc flags are active

Variables	Descripción
<i>modo de visualización</i>	Cambiar el modo de visualización de los registros, "d" significa que está configurado el valor predeterminado de fábrica
<i>umbral cargado</i>	Consumo de corriente en mA que el concentrador detecta que un dispositivo está completamente cargado, "d" significa que está configurado el valor predeterminado de fábrica
<i>temperatura de apagado</i>	Temperature that will shutdown the hub if reached in Celsius, "d" means factory default is set
<i>tambalear</i>	Un retraso entre los puertos que se encienden en ms, "d" significa que está configurado el valor predeterminado de fábrica

*Puerto de carga descendente (CDP) Estar habilitado significa que un puerto es capaz de transferir datos y cargar el dispositivo al mismo tiempo con una corriente más alta que solo la sincronización de datos. With CDP enabled the hub can supply up to 1.5 A

Si deshabilita CDP, recibirá la notificación "Este concentrador tiene el modo UCS del puerto descendente de carga deshabilitado. This could limit the maximum current seen on some ports." This notification is there to ensure you haven't turned this off by accident and can still have the highest charge available.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 5,
  "result": {
    "company_name": "cambrionix",
    "product_name": "SuperSync15",
    "local_name": "-",
    "attach_threshold": "d",
    "default_profile": [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
    "remap_ports": [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15],
    "ports_on": [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
    "sync_chrg": [1,1,1,1,1,1,1,1,1,1,1,1,1,1,1],
    "alt_sync_chrg": [1,1,1,1,1,1,1,1,1,1,1,1,1,1,1],
    "misc_flags": 0,
    "display_mode": "d",
    "charged_threshold": "d",
    "temperature_max": "d",
    "stagger": "d"
  }
}
```

Título del sistema

El texto de identificación del sistema.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Título del sistema"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": título del sistema
}
```

system-title es el nombre descriptivo completo del concentrador

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": "cambrionix U8S-EXT 8 Port USB Charge+Sync"
}
```

Temperatura_C

Temperatura actual de la PCB en grados Celsius.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Temperatura_C"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": temperatura
}
```

La *temperatura* es una temperatura medida como un decimal. ≤ 0 °C devolverá 0. Las temperaturas medidas ≥ 100 °C devolverán 100.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 37.7
}
```

Banderas_de_temperatura

Indicadores de error de temperatura:

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Temperature_flags"
  ]
}
```

manga de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": banderas de temperatura
}
```

indicadores de temperatura	Descripción
OT	se ha producido un evento de sobretemperatura.
vacío	la temperatura es aceptable

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": "OT"
}
```

Temperatura_límite_máx_C

Límite superior del rango de temperatura aceptable que activará el indicador de error.

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "Temperature_Limit_Max_C"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": temperatura-límite-máx.  
}
```

temperature-limit-max the upper limit in Celsius displayed as a decimal.

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": 65.0  
}
```

TemperaturaMax_C

Temperatura de PCB más alta registrada en grados Celsius.

Sintaxis: ver Call Structure

```
{  
  "método": "cbrx_connection_get",  
  "parámetros": [  
    mango de conexión ,  
    "TemperaturaMax_C"  
  ]  
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{  
  "resultado": temperatura-máx.  
}
```

La *temperatura* es una temperatura medida que se muestra como un decimal. ≤ 0 °C devolverá 0. Las temperaturas medidas ≥ 100 °C devolverán 100.

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": 39.9  
}
```

TotalCorriente_mA

Corriente total en mA para todos los puertos USB.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "CorrienteTotal_mA"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": corriente total
}
```

total-current es la corriente total en todos los puertos en mA que se muestra como un decimal

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 0
}
```

TotalPower_W

Potencia total consumida en todos los puertos USB en vatios (W).

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "PotenciaTotal_W"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones

```
{
  "result": "total-power"
}
```

total-power is the total power across all ports in watts displayed as an decimal.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 3.4
}
```

TwelveVoltRail_flags

Lista de indicadores de error del riel de suministro de 12 V.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "TwelveVoltRail_flags"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": banderas de doce voltios
}
```

banderas de doce voltios	Descripción
"ULTRAVIOLETA"	se produjo bajo voltaje
"VO"	se produjo sobretensión
"OV UV"	se produjeron tanto sobretensión como subtensión
" "	el voltaje es aceptable

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "result": " "  
}
```

TwelveVoltRail_Limit_Max_V

Límite superior del riel de 12V que activará la bandera de error.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "TwelveVoltRail_Limit_Max_V"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": límite máximo de doce voltios
}
```

doce-volt-limit-max es la cantidad máxima de voltios antes de que se marque el error en Volts mostrados como un decimal.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 14.5
}
```

TwelveVoltRail_Limit_Min_V

Límite inferior del riel de 12V que activará la bandera de error.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "TwelveVoltRail_Limit_Min_V"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": doce-volt-limit-min
}
```

doce-volt-limit-min es la cantidad mínima de voltios antes de que se marque el error en Voltios mostrados como un decimal.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 9.59
}
```

TwelveVoltRail_V

Tensión de alimentación actual de 12 V en voltios (V).

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "TwelveVoltRail_V"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": doce voltios
}
```

doce voltios es la cantidad de voltios que se suministran en voltios que se muestran como un decimal.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 12.43
}
```

TwelveVoltRailMax_V

Tensión de alimentación de 12 V más alta medida.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "TwelveVoltRailMax_V"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": doce voltios máx.
}
```

doce voltios máx. es la cantidad más alta de voltios que se ve en los voltios que se muestran como un decimal.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 12.52
}
```

TwelveVoltRailMin_V

Voltaje de suministro de 12 V más bajo medido en voltios (V).

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "TwelveVoltRailMin_V"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": doce voltios-min
}
```

doce voltios-mín. es la cantidad más pequeña de voltios que se ven en los voltios que se muestran como un decimal.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 12.31
}
```

Uptime_sec

Tiempo en segundos que el concentrador ha estado funcionando desde el último reinicio.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    mango de conexión ,
    "Uptime_sec"
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "resultado": tiempo de actividad
}
```

El tiempo de *actividad* es la cantidad de tiempo en segundos de ejecución continua, que se muestra como un número entero.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": 151304
}
```

14.3. Establecer diccionario

Llave	Conjunto de características
Pitido	5 V
ClearErrorFlags	base
ClearLCD	5 V
ClearRebootFlag	base
FiveVoltRail.Sobrevoltaje	5 V
FiveVoltRail.UnderVoltage	5 V
InputRail.Sobrevoltaje	PD
InputRail.bajo voltaje	PD
LCDText.ROW.COL	5 V
Mode	base
Port.N.puerta	puerta
Puerto.N.led1	base
Puerto.N.led2	base
Puerto.N.led3	base
Port.N.leds	base
Puerto.N.modos	base
Port.N.perfiles	sincronizar
Puerto.N.RGB	puerta
ProfileEnable.n	5 V
Reinicio	base
Control remoto	base
ControlRGB	puerta
SeguridadArmado	5 V

Llave	Conjunto de características
Ajustes	base
Temperatura.Exceso de temperatura	temperatura
TwelveVoltRail.Sobrevoltaje	12 V
TwelveVoltRail.UnderVoltage	12 V

Pitido

Pitido para el número de milisegundos transcurridos.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "bip",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
periodo de tiempo	Un número entero para la cantidad de tiempo requerida para el pitido en ms

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "beep",
    250
  ]
}
```

ClearErrorFlags

Borrar todos los indicadores de error

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "ClearErrorFlags",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Borrar las banderas de error

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "ClearErrorFlags",
    true
  ]
}
```

ClearLCD

Borre la pantalla LCD.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "ClearLCD",
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_set",
  "params": [
```

```
    7654,  
    "ClearLCD",  
  ]  
}
```

ClearRebootFlag

Borre la bandera de reinicio.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "ClearRebootFlag", Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Borrar la bandera Reiniciado

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "ClearRebootFlag",
    true
  ]
}
```

FiveVoltRail.Sobrevoltaje

Fuerce el comportamiento de una condición de sobrevoltaje de 5V.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "FiveVoltRail.OverVoltage", Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Establecer indicador de sobrevoltaje de 5V

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "FiveVoltRail.OverVoltage",
    true
  ]
}
```

FiveVoltRail.UnderVoltage

Forzar el comportamiento de una condición de bajo voltaje de 5V.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "FiveVoltRail.UnderVoltage", Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Establecer 5V bajo bandera de voltaje

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{  
  "jsonrpc": "2.0",  
  "id": 0,  
  "method": "cbrx_connection_set",  
  "params": [  
    7654,  
    "FiveVoltRail.UnderVoltage",  
    true  
  ]  
}
```

InputRail.Sobrevoltaje

Fuerce el comportamiento de una condición de sobrevoltaje del riel de entrada.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "InputRail.OverVoltage",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Establecer entrada de voltaje sobre el indicador de voltaje

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "InputRail.OverVoltage",
    true
  ]
}
```

InputRail.bajo voltaje

Forzar el comportamiento de un riel de entrada bajo condiciones de voltaje.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "InputRail.UnderVoltage",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Establecer entrada de voltaje bajo bandera de voltaje

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "LCDText.4.5",
    "hello"
  ]
}
```

LCDText.ROW.COL

Escriba la cadena en la pantalla LCD en (fila, columna). La fila y la columna están basadas en cero.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "LCDText.FILA.COL" ,
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>FILA</i>	Fila LCD que desea comenzar a escribir
<i>COLUMNA</i>	LCD Collum que deseas empezar a escribir
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
Cadena	Una cadena de texto que desea mostrar en la pantalla LCD

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "InputRail.UnderVoltage",
    true
  ]
}
```

Mode

Configure el mismo modo para todos los puertos USB. Consulte [los manuales de usuario del producto](#) para obtener detalles sobre los modos admitidos por cada concentrador.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "modo",
    " Valor "
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
c	modo de carga
s	Modo de sincronización y carga
b	modo sesgado
o	off

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "Mode",
    "s"
  ]
}
```

Port.N.puerta

Abre o cierra la puerta especificada. Debe monitorear el estado de la puerta requerida a través de `cbrx_connection_get(handle, "Gates")` para asegurarse de que se complete.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "Puerto.puerta N ",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>norte</i>	Número de puerto
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
abierto	Puerta abierta
cerca	Cerrar puerta
detener	Detener la acción actual de la puerta

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "Port.1.gate",
    "open"
  ]
}
```

Puerto.N.led1

Establecer el estado del primer LED

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "Puerto.N .led1",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>norte</i>	Número de puerto
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
Patrón de parpadeo	0-255 con los LED parpadeando según el patrón de bits representado por el valor.

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "Port.1.led1",
    170
  ]
}
```

Puerto.N.led2

Establecer el estado del segundo LED

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "Puerto.N .led2",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>norte</i>	Número de puerto
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
Patrón de parpadeo	0-255 con los LED parpadeando según el patrón de bits representado por el valor.

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "Port.1.led2",
    170
  ]
}
```

Puerto.N.led3

Establecer el estado del tercer LED

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "Puerto.N .led3",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>norte</i>	Número de puerto
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
Patrón de parpadeo	0-255 con los LED parpadeando según el patrón de bits representado por el valor.

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "Port.1.1ed3",
    170
  ]
}
```

Port.N.leds

Establecer el estado de los tres LED

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "Puerto.N .leds",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>norte</i>	Número de puerto
<i>Valor</i>	El valor que desea establecer para la clave

Valor

Un valor numérico de 24 bits que consiste en la configuración de LED individuales como valores de 8 bits desplazados y combinados con OR. es decir $led1 | (led2 \ll 8) | (led3 \ll 16)$, por lo que con $led1$ y $led2$ como cero, y siendo $led3$ $0b10101010$ (decimal 170), el resultado debería ser 11,141,120 decimal.

En un ThunderSync3, 255 es verde, 65 280 es rojo, 16 711 680 es amarillo.

En un ModIT, se usa azul en lugar de amarillo, pero, por supuesto, puede mezclar colores en cualquier mezcla RGB.

Devoluciones:

```
{
```

```
"resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "Port.1.1eds",
    11193404
  ]
}
```

Puerto.N.modos

Configure el modo de un solo puerto USB. El modo de sincronización solo se puede configurar en un dispositivo que implemente el conjunto de funciones de sincronización. El modo sesgado solo se puede configurar en dispositivos que implementan el conjunto de funciones de 5V.

Sintaxis: ver Call Structure

```

{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "Puerto.N .modo",
    Valor
  ]
}

```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>norte</i>	Número de puerto
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
c	modo de carga
s	Modo de sincronización y carga
b	modo sesgado
o	Apagado

Devoluciones:

```

{

```

```
"resultado": verdadero  
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{  
  "id": 0,  
  "jsonrpc": "2.0",  
  "method": "cbrx_connection_set",  
  "params": [  
    7654,  
    "Port.1.Mode",  
    "c"  
  ]  
}
```

Port.N.perfiles

Establezca la lista de perfiles habilitados.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "Puerto.N .perfiles",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>norte</i>	Número de puerto
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
perfil	Una lista de perfiles separados por comas para habilitar, consulte los manuales de usuario del producto para obtener detalles sobre los perfiles aplicables a su concentrador.

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "Port.1.profiles",
    "1,2,3"
  ]
}
```

Puerto.N.RGB

Establezca el color RGB de los LED ModIT.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "Puerto.N .RGB",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>norte</i>	Número de puerto
<i>Valor</i>	El valor que desea establecer para la clave

Valor

El valor del color puede ser un número entero (donde debe proporcionar RGBA completo) o una cadena. Para una cadena, puede especificarla como RGB, RGBA, RRGGBB o RRGGBBAA. Al igual que puede hacerlo con un color HTML. Por ejemplo, utilice "FF0000" o "F00" para rojo, "FFFFFF" para blanco, etc. Opcionalmente proporcione los dígitos alfabéticos (intensidad), por lo que "FFFFFF80" para la mitad de blanco brillante.

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "Port.1.RGB",
    "ff08"
  ]
}
```

ProfileEnable.n

Habilitar o deshabilitar el perfil global n

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "ProfileEnable. n "
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>norte</i>	Número de perfil

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
```

```
"method": "cbrx_connection_set",  
"params": [  
    7654,  
    "ProfileEnable.n"  
]  
}
```

Reinicio

Reinicie el concentrador ahora. La API intentará restablecer la conexión automáticamente, pero no debe esperar recibir resultados actualizados durante varios segundos.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "Reiniciar",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Reinicie el concentrador.

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "Reboot",
    "true"
  ]
}
```

Control remoto

Activar/desactivar el control de los controles de la unidad. Esto permitirá que los LED o la pantalla LCD se actualicen o que se detecten las pulsaciones de los botones del panel.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "Control remoto",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Habilitar el modo de control remoto manual
FALSO	Deshabilitar el modo de control remoto manual
"auto"	Habilite el modo de control automático a través de la API

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "RemoteControl",
    "true"
  ]
}
```

ControlRGB

Habilitar/deshabilitar el control LED ModIT RGB para puertos. Esto no requiere que RemoteControl esté habilitado.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "RGBControl",
    {
      "puerto": N ,
      "habilitar": valor
    }
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>norte</i>	número de puerto
<i>valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Habilitar el control de los LED RGB
FALSO	Deshabilitar el control de los LED RGB

Devoluciones:

```
{
```

```
"resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "RGBControl",
    {
      "port": 8,
      "enable": true
    }
  ]
}
```

Múltiples puertos

Si desea establecer el control en un rango de puertos, los parámetros cambiarán. Deberá ingresar dos valores, el valor 'inicio' del puerto para comenzar y el valor 'final' del puerto para terminar.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "RGBControl",
    { "inicio": N , "fin": N , "habilitar": valor
      }
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>norte</i>	número de puerto
<i>valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Habilitar el control de los LED RGB
FALSO	Deshabilitar el control de los LED RGB

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "RGBControl",
    {
      "start": 1,
      "end": 8,
      "enable": true
    }
  ]
}
```

SeguridadArmado

Activar/desactivar función de seguridad. Si la seguridad está habilitada, la eliminación de un dispositivo de un puerto hará sonar una alarma (si está instalada) y luces intermitentes (si están instaladas).

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "Seguridad Armada",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	habilitar la seguridad
FALSO	desactivar seguridad

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "SecurityArmed",
    "true"
  ]
}
```

Ajustes

Realice la operación de la interfaz de línea de comandos en el concentrador conectado y devuelva el resultado completo. Esto le permite ejecutar comandos directamente en la línea de comandos del concentrador sin detener el servicio API. In order to update the settings the CLI setting will require a settings_unlock\n prefix to the command for more information on using CLI commands please see the CLI documentation <https://www.cambrionix.com/cambrionix-cli>.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "Ajustes",
    "settings_unlock\nCommand"
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Comando</i>	The Command you wish to send to the hub For all CLI commands see the CLI Documentation https://www.cambrionix.com/cambrionix-cli

Devoluciones:

```
{
  "jsonrpc": "2.0",
  "identificación": 0,
  "result": "Unlocked \nSetting updated"
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

If you attempt to update a setting that is already set you will receive a message stating that the setting is already set such as the example of sending the ports on below

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "result": "Unlocked \nForcing ports on has already been defined."
}
```

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_hub_set",
  "params": [
    "DM01K2A8",
    "settings",
    "settings_unlock\nsettings_set ports_on 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1"
  ]
}
```

Temperatura.Exceso de temperatura

Forzar el comportamiento de una condición de sobret temperatura.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    mango de conexión ,
    "Temperatura.Exceso de temperatura",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Establecer el indicador de temperatura excesiva

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "Temperature.OverTemperature",
    "true"
  ]
}
```

TwelveVoltRail.Sobrevoltaje

Fuerce el comportamiento de una condición de sobrevoltaje de 12V. TwelveVoltRail es lo mismo que InputRail.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "TwelveVoltRail.OverVoltage",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Establecer el indicador de sobrevoltaje de 12 V

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "TwelveVoltRail.OverVoltage",
    "true"
  ]
}
```

TwelveVoltRail.UnderVoltage

Forzar el comportamiento de una condición de bajo voltaje de 12V.

TwelveVoltRail es lo mismo que InputRail.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_set",
  "parámetros": [
    identificador de conexión , "TwelveVoltRail.UnderVoltage",
    Valor
  ]
}
```

Parámetro	Descripción
<i>empuñadura de conexión</i>	El Connection Handles como un entero
<i>Valor</i>	El valor que desea establecer para la clave

Valor	Descripción
verdadero	Establezca la bandera de bajo voltaje de 12V

Devoluciones:

```
{
  "resultado": verdadero
}
```

errores

Si hay un error en el método API entonces un [JSON-RPC Error Object](#) Será devuelto.

Ejemplo

```
{
  "id": 0,
  "jsonrpc": "2.0",
  "method": "cbrx_connection_set",
  "params": [
    7654,
    "TwelveVoltRail.UnderVoltage",
    "true"
  ]
}
```

14.4. Diccionarios obsoletos

Estos diccionarios existen solo para admitir la compatibilidad con versiones anteriores y no deben usarse. Estos pares clave-valor pueden eliminarse en futuras versiones.

Llamada API	Descripción
Ajustes	Obtain current hub Internal hub settings.

Ajustes

Obtain current hub Internal hub settings, returns as text.

Sintaxis: ver Call Structure

```
{
  "método": "cbrx_connection_get",
  "parámetros": [
    identificador de conexión , "Configuración",
  ]
}
```

manija de conexión es el [Connection Handles](#) como un número entero.

Devoluciones:

```
{
  "jsonrpc": "2.0",
  "identificación": 5,
  "resultado": [
    "Configuración de memoria actual:",
    "",
    "settings_set company_name nombre del fabricante ",
    "settings_set nombre_producto nombre-producto ",
    "settings_set nombre_local nombre-local ",
    "settings_set Attach_Threshold Attach - Threshold",
    "settings_set perfil_predeterminado perfil-predeterminado ",
    "settings_set remap_ports orden de puertos ",
    "settings_set ports_on puertos-on ",
    "settings_set sync_chrg carga de sincronización ",
  ]
}
```

```

"settings_set alt_sync_chrg alt-sync-carga ",
"settings_set misc_flags Internal hub-flags",
"settings_set display_mode modo de visualización ",
"settings_set cargado_umbral cargado-umbral ",
"settings_set temperature_max shutdown-temperature ",
"settings_set escalonar escalonar "
]
}

```

Variables	Descripción
<i>Nombre del Fabricante</i>	Nombre definido del fabricante, por defecto es 'Cambrionix'
<i>nombre del producto</i>	Nombre de hardware del producto
<i>nombre local</i>	Local name set by the user, "-" if not set
<i>adjuntar-umbral</i>	Consumo de corriente en mA que el concentrador detecta que un dispositivo está conectado, "d" significa que el valor predeterminado de fábrica está configurado
<i>perfil por defecto</i>	Perfil predeterminado para cada puerto
<i>orden de puerto</i>	Ordenar los puertos son por número de puerto
<i>puertos activados</i>	Si cada puerto está activado de forma predeterminada, 0 está desactivado de forma predeterminada 1 está activado de forma predeterminada
<i>sincronización de carga</i>	Si CDP* en cada puerto está activado, 0 está desactivado, 1 está activado, lista separada por comas
<i>alt-sync-carga</i>	Si CDP* alternativo en cada puerto está activado, 0 está desactivado, 1 está activado

Variables	Descripción
<i>Internal hub-flags</i>	If any Internal hub Misc flags are active
<i>modo de visualización</i>	Cambiar el modo de visualización de los registros, "d" significa que está configurado el valor predeterminado de fábrica
<i>umbral cargado</i>	Consumo de corriente en mA que el concentrador detecta que un dispositivo está completamente cargado, "d" significa que está configurado el valor predeterminado de fábrica
<i>temperatura de apagado</i>	Temperature that will shutdown the hub if reached in Celsius, "d" means factory default is set
<i>tambalear</i>	Un retraso entre los puertos que se encienden en ms, "d" significa que está configurado el valor predeterminado de fábrica

*Puerto de carga descendente (CDP) Estar habilitado significa que un puerto es capaz de transferir datos y cargar el dispositivo al mismo tiempo con una corriente más alta que solo la sincronización de datos. With CDP enabled the hub can supply up to 1.5 A

Si deshabilita CDP, recibirá la notificación "Este concentrador tiene el modo UCS del puerto descendente de carga deshabilitado. This could limit the maximum current seen on some ports." This notification is there to ensure you haven't turned this off by accident and can still have the highest charge available.

Ejemplo

```
{
  "jsonrpc": "2.0",
  "id": 5,
  "result": [
    "Current memory Settings :",
    "",
    "settings_set company_name cambrionix",
    "settings_set product_name SuperSync15",
    "settings_set local_name -",
    "settings_set attach_threshold d",
    "settings_set default_profile 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ",
    "settings_set remap_ports 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 ",
    "settings_set ports_on 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0",
    "settings_set sync_chrg 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1",
    "settings_set alt_sync_chrg 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 ",
    "settings_set misc_flags 0000",
    "settings_set display_mode d",
  ]
}
```

```
        "settings_set charged_threshold d",  
        "settings_set temperature_max d",  
        "settings_set stagger d"  
    ]  
}
```

15. Conexiones de enchufe

Cuando se utiliza el envoltorio de Python que proporciona el módulo `cbrxapi`, cada vez que se realiza una llamada a la API, se crea un socket. Este socket se usa para enviar el comando y recibir la respuesta antes de cerrarse.

Si está escribiendo su propio programa, es posible que desee considerar la creación de un solo socket al comienzo de su comunicación con la API y mantener este socket abierto hasta que desee dejar de usar la API. Mantener el socket abierto durante la vida útil de su comunicación con la API reducirá la carga en el sistema y conducirá a ciclos de comunicación más cortos con la API.

Si elige administrar sus propias conexiones de socket a la API, es importante que no cierre el socket antes de recibir la respuesta del comando final. Cerrar el socket sin esperar a recibir la respuesta puede hacer que la operación solicitada no se complete, esto es especialmente importante en las operaciones de ajuste y cierre.

16. Control de los LED

La API puede controlar los LED del producto. De forma predeterminada, el producto controla automáticamente estos LED para indicar el estado en el que se encuentra un puerto.

Para que los LED sean controlados por la API, este control automático debe estar deshabilitado y esto se hace configurando la tecla "RemoteControl" en 'True'. Si desea devolver el control de los LED al control automático, establezca "RemoteControl" en 'False'. Ver [conjunto_conexión_cbrx](#) para obtener más información sobre el uso de este método.

El control de un LED se logra proporcionando un valor de 8 bits que se interpreta en binario como un patrón que se repite continuamente. Entonces, al establecer el valor 11110000b, el LED parpadeará lentamente. El LED estará encendido donde haya un '1' y apagado donde haya un '0'. Alternativamente, establecer el valor 10101010b hará que el LED parpadee rápidamente. No es necesario que el patrón sea simétrico, por lo que 10010000b producirá dos breves destellos juntos con una pausa más larga antes de que se repita el ciclo.

Cualquier valor establecido para un LED mientras RemoteControl es False se sobrescribirá y, por lo tanto, no tendrá ningún efecto.

Un argumento especial de "auto" en lugar de True permite que el concentrador anule el patrón de LED establecido por el usuario cuando se quita un dispositivo conectado a ese puerto.

17. Información de la batería

La información de la batería se puede recuperar para los dispositivos conectados. Para dispositivos Android™ que usan Android Debug Bridge (ADB) y para dispositivos iOS, una compilación integrada de libimobile.

ADB se puede utilizar para consultar el nivel de batería en cualquier dispositivo Android™ siempre que se cumplan algunas condiciones.

- Las herramientas de la plataforma Android están instaladas, estas se pueden descargar desde [aquí](#).
- El binario ADB está en la ruta, o su ruta se proporciona a la API a través de `cbrx_config_set`.
- El dispositivo tiene habilitada la depuración USB.
- Ha confiado en la computadora desde el teléfono si el teléfono lo requiere.

Consulte [esta página](#) para obtener detalles sobre cómo habilitar el modo de depuración en dispositivos Android™. Las únicas opciones que se requieren son habilitar el modo de desarrollador y la depuración de USB.

```
# Install Android platform tools on Linux sudo apt install
Android-platform-tools# # Install Android platform tools on macOS
brew cask install Android-platform-tools # Install Android
platform tools on Windows # Goto
https://developer.Android.com/studio/releases/platform-tools # Download
SDK Platform-Tools for Windows # Extract and add the folder to your
path
or use # cbrx_config_set("adb_path" <pathname>) to add to API
settings.
```

Encontrar adb sin la ruta

Alternativamente, al configurar la ruta, podemos decirle a la API dónde encontrar estos programas.

```
{
  "jsonrpc": "2.0",
  "id": 0,
  "method": "cbrx_config_set",
  "params": {
    "adb_path": "/usr/local/bin"
  }
}
```

Mobile-device battery trust-levels

To obtain the battery information on mobile devices (phones / tablets) the device must be paired with the host system. To pair a device you will need to trust the host system on the mobile device when first connecting. There are various trust levels which are documented below.

Trust level	Descripción
"not-paired"	Device not paired
"paired"	Device paired
"pending"	Battery information pending
"failed"	Failed to obtain battery information
"prohibited"	Prohibited from obtaining battery information
"error"	Error on obtaining battery information

18. API Error codes

Código	Valor	Descripción
CBRXAPI_CÓDIGO DE ERROR_IDNOTFOUND	-10001	Identificación no encontrada. El ID de unidad pasado no representa un concentrador o se ha desconectado desde la última vez que se ejecutó el descubrimiento. Tenga en cuenta que hay un tiempo de espera interno que cerrará los identificadores no utilizados después de un minuto.
CBRXAPI_CÓDIGO DE ERROR_NOHANDLINGTHREAD	-10002	No se puede comenzar a manejar el subproceso. Este error no es aplicable más allá de 2.1.
CBRXAPI_CÓDIGO DE ERROR_KEYNOTFOUND	-10003	Clave no encontrada. No se puede encontrar una clave pasada. Puede estar mal escrito o no existir en el diccionario para esta unidad.
CBRXAPI_CÓDIGO DE ERROR_VALOR DE CONFIGURACIÓN DE ERROR	-10004	No se pudo establecer el valor. El par (valor clave) no era aceptable. Esto podría significar que la clave no existe o está mal escrita, el valor es del tipo incorrecto o el valor pasado no es válido o está fuera de rango.
CBRXAPI_CÓDIGO DE ERROR_INVALIDHANDLE	-10005	Identificador no válido. El identificador pasado a una función no es válido o ya no es válido. Esto podría suceder ya sea al pasar un valor incorrecto o si el identificador ya se ha cerrado (es decir, por <code>cbx_closeandlock</code> siendo llamado) o la unidad ha sido desconectada de la computadora.
CBRXAPI_ERRORCODE_TIMEOUT	-10006	Tiempo de espera en la comunicación. Una operación hacia un centro tardó demasiado en completarse. Es posible que se haya desconectado o simplemente haya tardado en responder. Vale la pena volver a intentar la operación.
CBRXAPI_CÓDIGO DE ERROR_DROPPED	-10007	Se ha interrumpido la conexión del socket al control remoto.
CBRXAPI_CÓDIGO DE ERROR_METHOD_REMOVED	-10008	El método ha sido eliminado.

Código	Valor	Descripción
CBRXAPI_ERRORCODE_AGAIN	-10009	El sistema no está listo. Intentar otra vez. Es probable que esto se deba a una llamada muy rápida a una función API y el sistema no ha progresado lo suficiente durante el inicio para repararlo.
CBRXAPI_CÓDIGO DE ERROR_FIRMWARE_ACTUALIZACIÓN	-10010	Error al realizar la actualización del firmware.
CBRXAPI_CÓDIGO DE ERROR_FIRMWARE_FILE	-10011	Error de archivo de firmware. Esto normalmente se debe a errores de formato de archivo.
CBRXAPI_CÓDIGO DE ERROR_DISPOSITIVO_NO_ENCONTRADO	-10012	Dispositivo no encontrado.
CBRXAPI_ERRORCODE_CONNECTION_ERROR	-10014	No se pudo abrir la conexión del puerto serie al concentrador.
CBRXAPI_CÓDIGO DE ERROR_HUB_NO_ENCONTRADO	-10013	Centro no encontrado.

Uso de marcas comerciales, marcas comerciales registradas y otros nombres y símbolos protegidos

Este manual puede hacer referencia a marcas comerciales, marcas registradas y otros nombres y/o símbolos protegidos de empresas de terceros no relacionadas de ninguna manera con Cambrionix. Cuando aparecen, estas referencias son sólo para fines ilustrativos y no representan una aprobación de un producto o servicio por parte de Cambrionix, ni una aprobación de los productos a los que se aplica este manual por parte de la empresa externa en cuestión.

Cambrionix reconoce por la presente que todas las marcas comerciales, marcas comerciales registradas, marcas de servicio y otros nombres y/o símbolos protegidos contenidos en este manual y los documentos relacionados son propiedad de sus respectivos titulares.

"Mac® y macOS® son marcas comerciales de Apple Inc., registradas en EE. UU. y otros países y regiones".

"Intel® y el logotipo de Intel son marcas comerciales de Intel Corporation o sus subsidiarias".

"Thunderbolt™ y el logotipo de Thunderbolt son marcas comerciales de Intel Corporation o sus subsidiarias".

"Android™ es una marca registrada de Google LLC"

"Chromebook™ es una marca registrada de Google LLC".

"iOS™ es una marca comercial o una marca comercial registrada de Apple Inc, en los EE. UU. y en otros países, y está

utilizado bajo licencia".

"Linux® es la marca registrada de Linus Torvalds en los EE. UU. y otros países"

"Microsoft™ y Microsoft Windows™ son marcas comerciales del grupo de empresas Microsoft."

"Cambrionix® y el logotipo son marcas comerciales de Cambrionix Limited".

Todas las marcas comerciales y marcas registradas mencionadas son reconocidas y respetadas como propiedad de sus respectivos propietarios.

Aviso importante sobre información protegida

Tenga en cuenta que ciertos componentes de la tecnología de Cambrionix se consideran propiedad intelectual (PI) protegida de Cambrionix. Específicamente:

- Código fuente: El código fuente de nuestro software es propietario y no se puede proporcionar.

- Métodos propietarios: Las descripciones detalladas e implementaciones de nuestros métodos propietarios también están protegidas.

Como tal, las solicitudes de acceso al código fuente u otra información protegida serán rechazadas respetuosamente. Agradecemos su comprensión y cooperación.

Patentes Cambrionix

Título	Enlace	Numero de aplicacion	Conceder número
Puerto de sincronización y carga	GB2489429	1105081.2	2489429
CAMBRIONIX	ES00002646615	2646615	00002646615
CAMBRIONIX MUY INTELIGENTE...	ES00002646617	2646617	00002646617

Licencia

El uso de Cambrionix Hub API está sujeto a las condiciones de Cambrionix Connect SaaS, el documento se puede descargar y visualizar a través del siguiente enlace.

<https://downloads.cambrionix.com/documentation/en/Cambrionix-Connect-SaaS-Conditions.pdf>

El uso de Cambrionix Hub API está sujeto al acuerdo de Licencia de Cambrionix, el documento se puede descargar y visualizar mediante el siguiente enlace.

<https://downloads.cambrionix.com/documentation/en/Cambrionix-Licence-Agreement.pdf>

Cambrionix limitada
The Maurice Wilkes Building
Cowley Road
Cambridge CB4 0DS
Reino Unido

+44 (0) 1223 755520

<https://www.cambrionix.com>

Cambrionix Ltd es una empresa registrada en Inglaterra y Gales
con el número de empresa 06210854