

Multi IO V3

Manual do protocolo HTTP



Commbox
www.commbox.com.br

Sumário

Sumário.....	1
Lista de Abreviaturas.....	2
Introdução.....	3
Definições Gerais.....	4
Requisições.....	6
Resposta de erro.....	33
Contato.....	34

Lista de Abreviaturas

HTTP	Hypertext Transfer Protocol
ID	Identificador
IP	Internet Protocol
JSON	JavaScript Object Notation
ms	Milésimos de segundos
PHP	Hypertext Preprocessor
s	Segundos

Introdução

Este documento tem por objetivo descrever as informações necessárias para que desenvolvedores de software realizem a integração com os produtos da série MIO V3 através do protocolo HTTP.

Com a integração é possível realizar a execução de comandos na MIO V3 através de requisições HTTP, que são geradas a partir de URLs endereçadas ao IP do equipamento.

Para facilitar a demonstração das requisições e dos comandos descritos neste documento, foram criados exemplos que utilizam um navegador de internet (Chrome, Firefox, Internet Explorer e etc) para enviar as requisições e receber as respostas da MIO. Nesses exemplos, as requisições HTTP são originadas através do navegador onde, na sua barra de endereços, é informada a URL e os parâmetros para o acionamento de comandos na MIO. As respostas dos comandos são exibidas no corpo da página do navegador ao término da requisição.

Também são demonstrados exemplos de requisições HTTP utilizando a linguagem de programação PHP. Vale ressaltar que a integração pode ser realizada em qualquer linguagem de programação.

Este documento limita-se em descrever informações relacionadas a integração com a MIO V3. Portanto, não serão explicados conceitos sobre o protocolo HTTP e linguagens de programação.

Este documento é válido para os produtos da série MIO V3 com versão de firmware 3.02 ou superior.

Definições Gerais

Quantidade de entradas e relés:

A série MIO possui 5 modelos de placas, sendo que cada modelo possui uma quantidade específica de entradas e relés, conforme mostrado na tabela 1:

Modelo MIO	Quantidade de entradas	Quantidade de relés
MIO402	4	2
MIO400	4	4
MIO800	8	8
MIO2408	24	8
MIO0816	8	16

Tabela 1 – Modelo MIO x Quantidade de entradas e relés.

Em requisições que precisam indicar qual entrada ou relé deseja-se utilizar, a especificação da tabela 1 é utilizada como limite do valor a ser inserido.

Estado das entradas e dos relés:

No decorrer desse documento o estado das entradas e dos relés é representado por um valor numérico. Essa representação é mostrada na tabela 2.

Estado	Representação numérica
Entrada desatuada	0
Entrada atuada	1
Relé desligado	0
Relé ligado	1

Tabela 2 – Representação numérica do estado das entradas e dos relés.

Quando o comando para pulsar relé é enviada para a MIO, enquanto o relé estiver pulsando, a resposta para esse comando de ler o estado do relé será o mesmo valor do estado que foi passado no comando para pulsar o relé. Por exemplo, se um comando para pulsar 10 vezes o relé 1 é enviada para a MIO, com o estado do relé igual a ligado, então enquanto a MIO estiver pulsando o relé 1 essas 10 vezes o resultado para um comando de ler o estado do relé 1 será igual a relé ligado; após os 10 pulsos do relé 1 o resultado para um comando de ler o estado do relé 1 será igual a relé desligado.

Endereço IP:

Por padrão de fábrica a MIO é configurada com o endereço IP 192.168.0.100.

Porta HTTP:

Por padrão de fábrica a MIO é configurada com a porta 80 para as comunicações ethernet via protocolo HTTP.

A porta padrão do protocolo HTTP é a 80. Nos navegadores de internet, quando se faz uma comunicação via protocolo HTTP pela sua porta padrão (porta 80) a mesma pode ser omitida na URL.

Base de tempo da MIO:

Os tempos relacionados as requisições de atuar sobre os relés (set_output) e configuração das entradas (set_input_config e get_input_config) são inseridos na base de tempo de 1ms, ou seja, a inserção de um valor de 100 em uma dessas requisições se refere ao tempo de 100ms. Contudo, a MIO conta o tempo na base de 10ms, sendo assim, alguns valores podem sofrer arredondamentos, ou seja, valores que não sejam múltiplos de 10ms são alterados da seguinte forma:

- valores de 1ms até 9ms são alterados para 10ms;
- valores maiores que 10ms, que não são múltiplos de 10ms, são alterados para o valor menor mais próximo e que é múltiplo de 10ms, ou seja, ele é arredondado para baixo;

A tabela 3 mostra alguns exemplos de valores inserido no comando e seus valores adotados pela MIO.

Valor inserido	Valor adotado
1ms	10ms
5ms	10ms
59ms	50ms
127ms	120ms
2991ms	2990ms

Tabela 3 – Base de Tempo – Valor inserido x Valor adotado.

Formatação de texto dos exemplos de resposta:

Neste documento, os exemplos de respostas que a MIO retorna para as requisições enviadas são exibidas em uma formatação de texto para melhor visualização e entendimento. No contexto prático, a MIO envia as respostas sem nenhuma formatação de texto, como quebras de linha ou espaços em branco entre os caracteres da resposta.

Todas as respostas que a MIO envia estão no formato JSON.

Relógio interno MIO:

Observação: As requisições relacionadas ao relógio interno da MIO não são válidas para o modelo MIO402, uma vez que este recurso não está disponível neste modelo.

O relógio interno da MIO funciona no formato 24 horas (de 0 a 23 horas).

Requisições

Formatação de URLs:

Nos exemplos abaixo são demonstradas a construção de URLs, onde cada campo está demarcado por colchetes:

- 1 - Exemplo de URL de requisição sem porta especificada, ou seja, usa a porta 80 (padrão do HTTP):

http://[endereço IP]/[comando]?[parâmetro 1]=[valor do parâmetro 1]&[parâmetro n]=[valor do parâmetro n].

- 2 - Exemplo de URL de requisição com porta especificada:

http://[endereço IP]:[porta]/[comando]?[parâmetro 1]=[valor do parâmetro 1]&[parâmetro n]=[valor do parâmetro n].

Definição dos campos da URL de requisição:

- **endereço IP:** é o endereço de IP da MIO que se deseja estabelecer a comunicação ethernet via protocolo HTTP.
- **porta:** é a porta que está configurada na MIO para a comunicação ethernet via protocolo HTTP.
- **comando:** é a função que se deseja executar na MIO. Os comandos disponíveis estão listados na tabela 4.

#	Comando	Descrição
1	set_output	Atuar sobre os relés.
2	get_input_status	Ler estado e transição das entradas.
3	get_output_status	Ler estado e transição dos relés.
4	get_io_status	Ler estado e transição das entradas e dos relés.
5	set_input_config	Configurar as entradas
6	get_input_config	Ler a configuração das entradas.
7	set_ip_config	Configurar os parâmetros de rede da MIO.
8	get_ip_config	Ler os parâmetros de rede da MIO.
9	get_device_info	Ler informações sobre a MIO.
10	get_rtc	Ler o relógio interno da MIO.
11	set_rtc	Configurar o relógio interno da MIO.

Tabela 4 – Comandos.

- **parâmetro:** alguns comandos necessitam de parâmetros para especificar, por exemplo, qual relé deve ser acionado na função set_output ou ainda se o relé

deve ser ligado ou desligado. Os parâmetros podem ser obrigatórios ou opcionais.

- **valor do parâmetro:** quando for utilizado um parâmetro, então é obrigatório informar o seu valor. Cada parâmetro possui uma faixa de valor aceitável.

Resposta da MIO:

A MIO sempre retorna uma resposta com o resultado da execução do comando, sendo que a resposta está no formato JSON.

A resposta enviada pela MIO é formada pelos campos *result* e *data*.

O campo *result* pode ter as seguintes informações: *success* ou *error*. E o campo *data* pode conter as informações que foram solicitadas pelo comando ou não conter nenhuma informação.

Existem três possibilidades de resposta:

- ➔ Resposta para confirmação de comando executado com sucesso, sem informações adicionais:

```
{
  "result": "sucess",
  "data": null
}
```

- ➔ Resposta para confirmação de comando executado com sucesso, com informações adicionais:

```
{
  "result": "sucess",
  "data": [informações adicionais]
}
```

- ➔ Resposta informando que houve um erro na execução de um comando:

```
{
  "result": "error",
  "data": {
    "code": [código de erro],
    "code_message": "[mensagem de erro]",
    "code_data": [código do dado errado],
    "code_data_message": "[mensagem do dado errado]"
  }
}
```


Definição de comandos:

Para os exemplos posteriores o endereço de IP da MIO utilizado será o 192.168.0.100.

→ 1 - Atuar sobre os relés:

- **Função:** set_output.
- **Descrição:** Comando usado para atuar sobre um dos relés da MIO.
- **Parâmetros:**

#	Nome	Descrição	Valores aceitos	Obrigatório
1	address	Relé a ser atuado.	1 ao nº máximo de relés da MIO (ver tabela 1)	Sim
2	state	Estado do relé.	0 = relé desligado 1 = relé ligado 2 = inverter estado do relé	Sim
3	time_1	Tempo que o relé permanece no estado definido pelo parâmetro state.	0 a 9999999 (ms)	Não
4	time_2	Tempo que o relé permanece no estado inverso ao definido pelo parâmetro state.	0 a 9999999 (ms)	Não
5	n_cycles	Quantidade de vezes que o relé pulsará.	1 a 999	Não
6	time_interval	Tempo que o relé permanece no estado inverso ao definido pelo parâmetro state, antes de repetir o ciclo de pulsos.	0 a 9999999 (ms)	Não
7	time_total	Tempo total que o relé ficará pulsando.	0 a 9999999 (ms)	Não

Tabela 5 – Parâmetros do comando set_output.

Através dos parâmetros relacionados na tabela 5 é possível gerar ações diferentes para os relés. As possíveis ações que podem ser geradas e a relação de parâmetros utilizados são mostradas na tabela 6. Qualquer comando de atuar sobre um relé que possuía uma combinação de

parâmetros diferente das previstas na tabela 6 é considerado como um comando inválido e não é executado pelo MIO.

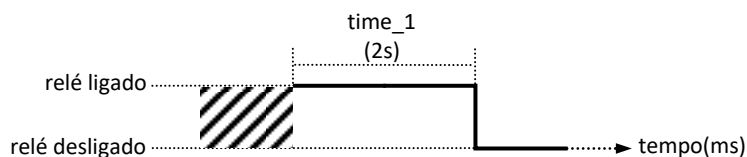
#	Ação do relé	Parâmetros utilizados
1	Definir um estado para o relé assumir: ligado, desligado ou inverter o estado atual.	address state
2	Gerar um pulso.	address state time_1
3	Repetir um pulso por tempo indeterminado.	address state time_1 time_2
4	Repetir um pulso pela quantidade de vezes definida.	address state time_2 n_cycles
5	Repetir um pulso por um tempo definido.	address state time_2 time_total
6	Repetir um ciclo de pulsos, com intervalo, por tempo indeterminado.	address state time_2 n_cycles time_interval
7	Repetir um ciclo de pulsos, com intervalo, por um tempo definido.	address state time_1 time_2 n_cycles time_interval time_total

Tabela 6 – Ação do relé x Parâmetros utilizados.

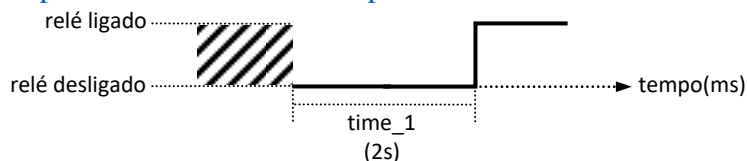
Nos comandos que o parâmetro time_total é utilizado, o valor dele deve ser igual ou maior que a soma dos valores dos parâmetros de tempo usados nesse mesmo comando.

- **Exemplos do comando:**

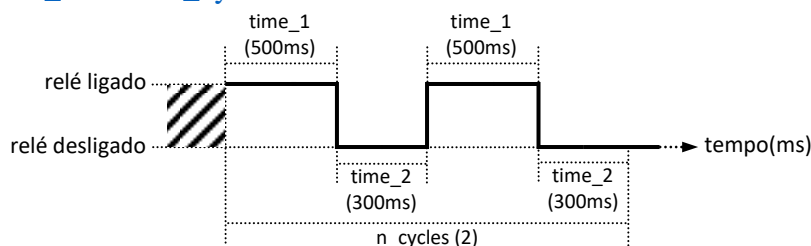
- Ligar o relé 1:
http://192.168.0.100/set_output?address=1&state=1.
- Desligar o relé 1:
http://192.168.0.100/set_output?address=1&state=0.
- Ligar o relé 3 por 2s:
http://192.168.0.100/set_output?address=3&state=1&time_1=2000.



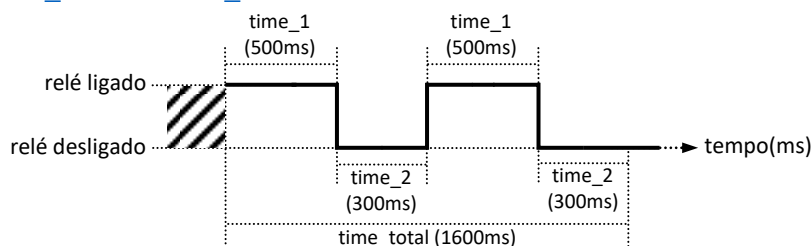
- Desligar o relé 3 por 2s:
http://192.168.0.100/set_output?address=3&state=0&time_1=2000.



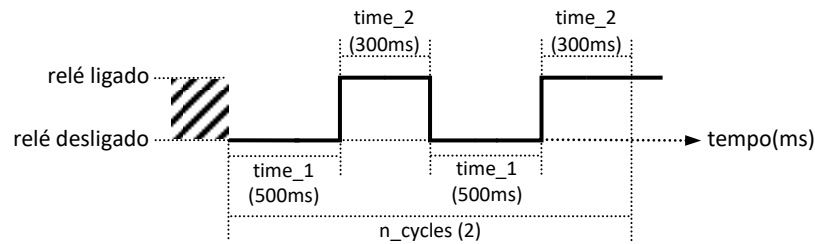
- Pulsar o relé 4, permanecendo ligado por 500ms e desligado por 300ms. Repete esse pulso 2 vezes (utiliza o parâmetro n_cycles):
http://192.168.0.100/set_output?address=4&state=1&time_1=500&time_2=300&n_cycles=2.



- Pulsar o relé 4, permanecendo ligado por 500ms e desligado por 300ms. Repete esse pulso 2 vezes (utiliza o parâmetro time_total):
http://192.168.0.100/set_output?address=4&state=1&time_1=500&time_2=300&time_total=1600.



- Pulsar o relé 4, permanecendo desligado por 500ms e ligado por 300ms. Repete esse pulso 2 vezes (utiliza o parâmetro n_cycles):
http://192.168.0.100/set_output?address=4&state=0&time_1=500&time_2=300&n_cycles=2.

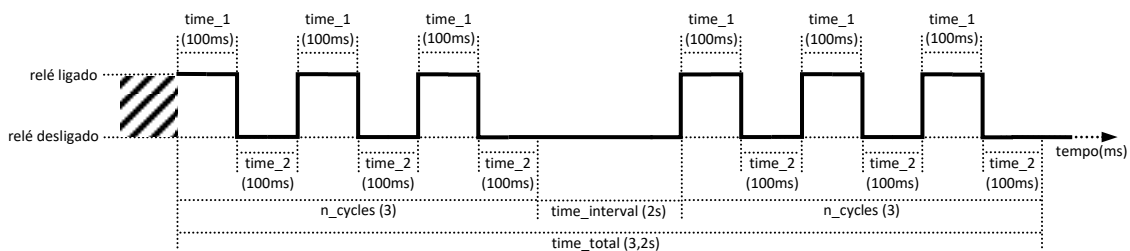


- Pulsar o relé 1, permanecendo ligado por 2s e desligado por 1s. Repete esse ciclo por um tempo indeterminado:

http://192.168.0.100/set_output?address=1&state=1&time_1=2000&time_2=1000.

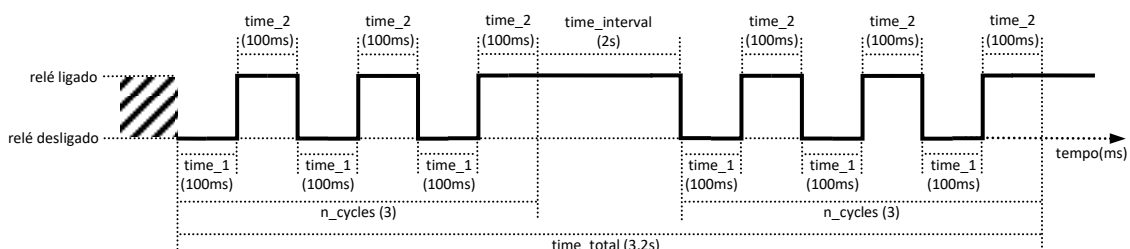
- Pulsar 3 vezes o relé 1, permanecendo ligado por 100ms, desligado por 100ms e com um intervalo de 2s entre os ciclos de pulsos. Repete esse ciclo por um tempo de 3,2s:

http://192.168.0.100/set_output?address=1&state=1&time_1=100&time_2=100&n_cycles=3&time_interval=2000&time_total=3200.



- Pulsar 3 vezes o relé 1, permanecendo desligado por 100ms, ligado por 100ms e com um intervalo de 2s entre os ciclos de pulsos. Repete esse ciclo por um tempo de 3,2s:

http://192.168.0.100/set_output?address=1&state=0&time_1=100&time_2=100&n_cycles=3&time_interval=2000&time_total=3200.



- Pulsar 3 vezes o relé 1, permanecendo ligado por 100ms, desligado por 100ms e com um intervalo de 2s entre os ciclos de pulsos. Repete esse ciclo por um tempo indeterminado:

http://192.168.0.100/set_output?address=1&state=1&time_1=100&time_2=100&n_cycles=3&time_interval=2000.

- **Resposta do comando:**
A MIO envia uma resposta de confirmação que o comando foi executado, sem informações adicionais.
- **Exemplo de código PHP para o comando:**
Pulsar o relé 4 duas vezes, permanecendo ligado por 500ms e desligado por 300ms.

```
$param = "address=1&state=1&time_1=500&time_2=300&time_total=1600";  
  
$url = "http://192.168.0.100/set_output?". $param;  
  
$ch = curl_init ();  
  
curl_setopt ($ch, CURLOPT_URL, $url);  
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYPEER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYHOST, 2);  
curl_setopt ($ch, CURLOPT_HTTPHEADER, array ('Accept: application/json'));  
  
$response = curl_exec ($ch);  
curl_close ($ch);  
  
echo $response;
```

→ 2 - Ler estado e transição das entradas:

- **Comando:** `get_input_status`.
- **Descrição:** Comando usado para ler o estado de todas as entradas da MIO e se ocorreu alguma transição no estado das entradas desde a última leitura.
- **Parâmetros:**

#	Nome	Descrição	Valores aceitos	Obrigatório
1	format	Formato da resposta da MIO para esse comando. Se o parâmetro não for utilizado, a MIO responderá no formato campos.	0 = formato campos 1 = formato decimal 2 = formato binário	Não

Tabela 7 – Parâmetros do comando `get_input_status`.

- **Formato campos:** 32 números decimais, separados por vírgulas, são usados para representar as entradas. Cada número decimal representa uma entrada, sendo que o número mais a direita representa a entrada 1 e assim sucessivamente.
- **Formato decimal:** apenas 1 número decimal representa todas as entradas. Cada entrada tem um peso na composição desse valor, esse peso é definido por: $(2^{(i-1)}) * (k)$; onde i é o número da entrada, por exemplo, para a entrada 3 o valor de i é 3; e k é o estado da entrada, sendo $k=0$ para a entrada desatuada ou sem transição e $k=1$ para entrada atuada ou com transição. O número decimal que representa todas as entradas é o somatório do peso de todas as entradas da MIO.
- **Formato binário:** 32 caracteres são usados para representar todas as entradas. Cada caractere representa uma entrada, sendo que o caractere mais a direita representa a entrada 1 e assim sucessivamente.
- **Exemplos do comando:**
Para os exemplos abaixo é considerado que as entradas 1 e 4 estão atuadas e o restante das entradas estão desatuadas:
 - 1 - Ler o estado das entradas sem enviar o parâmetro `format`, ou seja, no formato campos:
http://192.168.0.100/get_input_status.
 - 2 - Ler o estado das entradas no formato decimal:
http://192.168.0.100/get_input_status?format=1.
 - 3 - Ler o estado das entradas no formato binário:
http://192.168.0.100/get_input_status?format=2.
- **Resposta do comando (de acordo com os exemplos acima):**


```
$url = "http://192.168.0.100/get_input_status";

$ch = curl_init();

curl_setopt ($ch, CURLOPT_URL, $url);
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt ($ch, CURLOPT_SSL_VERIFYPEER, true);
curl_setopt ($ch, CURLOPT_SSL_VERIFYHOST, 2);
curl_setopt ($ch, CURLOPT_HTTPHEADER, array ('Accept: application/json'));

$response = curl_exec ($ch);
curl_close ($ch);

echo $response;
```


→ 5 - Configurar as entradas:

- **Comando:** set_input_config.
- **Descrição:** Comando usado para configurar as entradas da MIO.
- **Parâmetros:**

#	Nome	Descrição	Valores aceitos	Obrigatório
1	address	Entrada a ser configurada.	1 ao n° máximo de entradas da MIO (ver tabela 1)	Sim
2	enable	Habilita ou desabilita a entrada.	0 = entrada desabilitada 1 = entrada habilitada	Sim
3	delay_on	Tempo que a entrada demora em assumir que seu estado mudou de desatuada para atuada.	0 a 999999 (ms)	Não
4	delay_off	Tempo que a entrada demora em assumir que seu estado mudou de atuada para desatuada.	0 a 999999 (ms)	Não

Tabela 8 – Parâmetros do comando set_input_config.

- **Observações:**

Parâmetro	Observações
enable	Uma entrada desabilitada sempre informa seu estado como desatuada. Por padrão de fábrica todas as entradas estão configuradas como habilitadas.
delay_on	Por padrão de fábrica esse tempo está configurado com 100ms.
delay_off	Por padrão de fábrica esse tempo está configurado com 100ms.

Tabela 9 – Observações do comando set_input_config.

- **Exemplos do comando:**

- Desabilitar a entrada 1:
http://192.168.0.100/set_input_config?address=1&enable=0.
- Habilitar a entrada 1:
http://192.168.0.100/set_input_config?address=1&enable=1.
- Alterar o delay_on para 1s e o delay_off para 1s:
http://192.168.0.100/set_input_config?address=1&enable=1&delay_on=1000&delay_off=1000.

- **Resposta do comando:**

A MIO envia uma resposta de confirmação que o comando foi executado, sem informações adicionais.

- **Exemplo de código PHP para o comando:**

Para a entrada 1, alterar o delay_on para 5s e o delay_off para 5s.

```
$param = "address=1&enable=1&delay_on=5000&delay_off=5000";  
  
$url = "http://192.168.0.100/set_input_config?". $param;  
  
$ch = curl_init ();  
  
curl_setopt ($ch, CURLOPT_URL, $url);  
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYPEER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYHOST, 2);  
curl_setopt ($ch, CURLOPT_HTTPHEADER, array ('Accept: application/json'));  
  
$response = curl_exec ($ch);  
curl_close ($ch);  
  
echo $response;
```

→ **6 – Ler a configuração das entradas:**

- **Comando:** `get_input_config`.
- **Descrição:** Comando usado para ler a configuração de uma das entradas da MIO.
- **Parâmetros:**

#	Nome	Descrição	Valores aceitos	Obrigatório
1	address	Entrada que será lida a configuração.	1 ao nº máximo de entradas da MIO (ver tabela 1)	Sim

Tabela 10 – Parâmetros do comando `get_input_config`.

- **Exemplos do comando:**
 - Ler a configuração da entrada 1:
http://192.168.0.100/get_input_config?address=1.
- **Resposta do comando (de acordo com o exemplo acima):**
 Considerando que a entrada 1 esteja habilitada, com o `delay_on=100ms` e com o `delay_off =100ms`.

```
{
  "result": "sucess",
  "data": {
    "address": 1,
    "enable": 1,
    "delay_on": 100,
    "delay_off": 100
  }
}
```

- **Exemplo de código PHP para o comando:**
 Ler as configurações da entrada 1.

```
$param = "address=1";

$url = "http://192.168.0.100/get_input_config?".$param;
$ch = curl_init ();

curl_setopt ($ch, CURLOPT_URL, $url);
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt ($ch, CURLOPT_SSL_VERIFYPEER, true);
curl_setopt ($ch, CURLOPT_SSL_VERIFYHOST, 2);
curl_setopt ($ch, CURLOPT_HTTPHEADER, array ('Accept: application/json'));
$response = curl_exec ($ch);
curl_close ($ch);
```



```
echo $response;
```

→ **7 - Configurar os parâmetros de rede da MIO:**

- **Comando:** set_ip_config.
- **Descrição:** Comando usado para configurar alguns parâmetros da rede ethernet da MIO. Após a execução desse comando a MIO será reinicializada para assumir os novos parâmetros configurados.
- **Parâmetros:**

#	Nome	Descrição	Valores aceitos	Obrigatório
1	ip	Endereço IP da MIO.	No formato: nnn.nnn.nnn.nnn onde nnn varia de 0 a 255.	Sim
2	port	Porta que a MIO utiliza para se comunicar via protocolo HTTP.	1 até 65535	Sim
3	mask	Sub máscara de rede da MIO.	No formato: nnn.nnn.nnn.nnn onde nnn varia de 0 a 255.	Sim
4	gateway	Endereço de IP do gateway.	No formato: nnn.nnn.nnn.nnn onde nnn varia de 0 a 255.	Sim

Tabela 11 – Parâmetros do comando set_ip_config.

- **Exemplos do comando:**
 - Alterar o endereço de IP da MIO para 192.168.0.101, a porta para 5091, a sub máscara de rede para 255.255.255.0 e o endereço de IP do gateway para 192.168.0.2:
http://192.168.0.100/set_ip_config?ip=192.168.0.101&port=5091&mask=255.255.255.0&gateway=192.168.0.2.
- **Resposta do comando:**

A MIO envia uma resposta de confirmação que o comando foi executado, sem informações adicionais.
- **Exemplo de código PHP para o comando:**

Alterar o endereço de IP da MIO para 192.168.0.100, a porta para 80, a sub máscara de rede para 255.255.255.0 e o endereço de IP do gateway para 192.168.0.1.

```
$param =  
"ip=192.168.0.100&port=80&mask=255.255.255.0&gateway=192.168.0.1";  
  
$url = "http://192.168.0.100/set_ip_config?".$param;  
  
$ch = curl_init ();  
  
curl_setopt ($ch, CURLOPT_URL, $url);  
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYPEER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYHOST, 2);  
curl_setopt ($ch, CURLOPT_HTTPHEADER, array ('Accept: application/json'));  
  
$response = curl_exec ($ch);  
curl_close ($ch);  
  
echo $response;
```

→ 8 - Ler os parâmetros de rede da MIO:

- **Comando:** get_ip_config.
- **Descrição:** Comando usado para ler os parâmetros da rede ethernet que a MIO está utilizando.
- **Parâmetros:** Este comando não possui parâmetros.
- **Exemplo do comando:** http://192.168.0.100/get_ip_config.
- **Resposta do comando:**

```
{  
  "result": "sucess",  
  "data": {  
    "source_ip": "192.168.0.100",  
    "source_port": 80,  
    "subnet_mask": "255.255.255.0",  
    "gateway_ip": "192.168.0.1",  
    "mac_address": "00-04-A3-3D-5F-91"  
  }  
}
```

- **Exemplo de código PHP para o comando:**

```
$url = "http://192.168.0.100/get_ip_config";  
  
$ch = curl_init ();  
  
curl_setopt ($ch, CURLOPT_URL, $url);  
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYPEER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYHOST, 2);  
curl_setopt ($ch, CURLOPT_HTTPHEADER, array ('Accept: application/json'));  
  
$response = curl_exec ($ch);  
curl_close ($ch);  
  
echo $response;
```

→ 9 - Ler informações sobre a MIO:

- **Comando:** get_device_info.
- **Descrição:** Comando usada para ler informações sobre a MIO, sendo elas: modelo da MIO, código do modelo, versão do firmware e ID único da MIO. A relação entre o modelo da MIO e o seu código está representado na tabela 12.

Modelo da MIO	Código do modelo
MIO402	40
MIO400	31
MIO800	32
MIO2408	34
MIO0816	41

Tabela 12 – Modelo MIO x Código do modelo.

- **Parâmetros:** Este comando não possui parâmetros.
- **Exemplo do comando:** http://192.168.0.100/get_device_info.
- **Resposta do comando:**

```
{  
  "result": "sucess",  
  "data": {  
    "model": "MIO800",  
    "model_code": 32,  
    "version_firmware": "3.02",  
    "unique_id": 2738708369  
  }  
}
```

- **Exemplo de código PHP para o comando:**

```
$url = "http://192.168.0.100/get_device_info";  
  
$ch = curl_init ();  
  
curl_setopt ($ch, CURLOPT_URL, $url);  
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYPEER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYHOST, 2);  
curl_setopt ($ch, CURLOPT_HTTPHEADER, array ('Accept: application/json'));  
  
$response = curl_exec ($ch);  
curl_close ($ch);  
  
echo $response;
```


→ 10 - Ler o relógio interno:

- **Comando:** get_rtc.
- **Descrição:** Comando usado para ler as informações de data e hora do relógio interno da MIO.
- **Parâmetros:** Este comando não possui parâmetros.
- **Exemplo do comando:** http://192.168.0.100/get_rtc.
- **Resposta do comando:**

- A data é retornada no formato yy-mm-dd, onde yy é o ano, mm é o mês e dd é o dia.

- A hora é retornada no formato hh:mm:ss, onde hh é a hora, mm são os minutos e ss são os segundos.

```
{  
  "result": "sucess",  
  "data": {  
    "date": "18-4-16",  
    "time": "11:57:35"  
  }  
}
```

- **Exemplo de código PHP para o comando:**

```
$url = "http://192.168.0.100/get_rtc";  
  
$ch = curl_init ();  
  
curl_setopt ($ch, CURLOPT_URL, $url);  
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYPEER, true);  
curl_setopt ($ch, CURLOPT_SSL_VERIFYHOST, 2);  
curl_setopt ($ch, CURLOPT_HTTPHEADER, array ('Accept: application/json'));  
  
$response = curl_exec ($ch);  
curl_close ($ch);  
  
echo $response;
```

→ **11 - Configurar o relógio interno:**

- **Comando:** set_rtc.
- **Descrição:** Comando usado para configurar o relógio interno da MIO.
- **Parâmetros:**

#	Nome	Descrição	Valores aceitos	Obrigatório
1	date	Define a data (ano, mês, dia) para o relógio interno da MIO.	No formato: yy-mm-dd onde yy é o ano, mm é o mês e dd é o dia.	Sim
2	time	Define o horário (hora, minutos, segundos) para o relógio interno da MIO.	No formato: hh:mm:ss onde hh é a hora, mm são os minutos e ss são os segundos.	Sim

Tabela 13 – Parâmetros do comando set_rtc.

- **Exemplos do comando:**
 - Alterar a data da MIO para 30 de janeiro de 2018 e o horário para 12 horas 30 minutos e 55 segundos:
http://192.168.0.100/set_rtc?date=18-01-30&time=12:30:55.
- **Resposta do comando:**
 A MIO envia uma resposta de confirmação que o comando foi executado, sem informações adicionais.
- **Exemplo de código PHP para o comando:**
 Alterar a data da MIO para 24 de abril de 2018 e o horário para 19 horas 5 minutos e 35 segundos.

```

$params = "date=18-04-24&time=19:5:35";
$url = "http://192.168.0.100/set_rtc?".$params;
$ch = curl_init ();
curl_setopt ($ch, CURLOPT_URL, $url);
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, true);
curl_setopt ($ch, CURLOPT_SSL_VERIFYPEER, true);
curl_setopt ($ch, CURLOPT_SSL_VERIFYHOST, 2);
curl_setopt ($ch, CURLOPT_HTTPHEADER, array ('Accept: application/json'));
$response = curl_exec ($ch);
curl_close ($ch);
echo $response;

```


Resumo dos comandos:

A tabela 14 mostra um resumo dos comandos, parâmetros e valores aceitáveis dos parâmetros.

#	Descrição	Comando	Parâmetros	
			Nome	Valores aceitáveis
1	Atuar sobre os relés	set_output	address	1 ao nº máximo de relés da MIO (ver tabela 1)
			state	0=desligado; 1=ligado; 2=inverter
			time_1	0 a 9999999 (ms)
			time_2	0 a 9999999 (ms)
			n_cycles	1 a 999
			time_interval	0 a 9999999 (ms)
			time_total	0 a 9999999 (ms)
2	Ler estado e transição das entradas	get_input_status	format	0=campos; 1=decimal; 2=binário
3	Ler estado e transição dos relés	get_output_status	format	0=campos; 1=decimal; 2=binário
4	Ler estado e transição das entradas e dos relés	get_ios_status	format	0=campos; 1=decimal; 2=binário
5	Configurar as entradas	set_input_config	address	1 ao nº máximo de entradas da MIO (ver tabela 1)
			enable	0=desabilita entrada 1=habilita entrada
			delay_on	0 a 9999999 (ms)
			delay_off	0 a 9999999 (ms)
6	Ler configuração das entradas	get_input_config	address	1 ao nº máximo de entradas da MIO (ver tabela 1)
7	Configurar os parâmetros de rede da MIO	set_ip_config	ip	nnn.nnn.nnn.nnn (nnn = 0 a 255)
			port	1 a 65535
			mask	nnn.nnn.nnn.nnn (nnn = 0 a 255)
			gateway	nnn.nnn.nnn.nnn (nnn = 0 a 255)
8	Ler os parâmetros de rede da MIO	get_ip_config		
9	Ler informações sobre a MIO	get_device_info		
10	Ler o relógio interno da MIO	get_rtc		
11	Configurar o relógio interno da MIO	set_rtc	date	yy-mm-dd (yy=ano; mm=mês; dd=dia)
			time	hh:mm:ss (hh=hora; mm=minuto; ss=segundos)

Tabela 14 – Comandos, parâmetros e valores aceitáveis.

Resposta de erro

Na resposta de erro, para o caso de algum erro na linha de comando, os possíveis códigos de erro e suas mensagens são mostrados na tabela 15. Os campos dessa tabela preenchidos com colchetes variam, pois mostram a mensagem recebida pela MIO e que são desconhecidos para a mesma.

code	code_message	code_data	code_data_message		
1	Method not allowed	null	null		
2	Unknown command	1	[comando recebido]		
3	Unknown parameter	1	[parâmetro recebido]		
4	Missing parameter	10	address		
		11	state		
		12	time_1		
		13	time_2		
		14	n_cycles		
		15	time_interval		
		16	time_total		
		17	format		
		5	Duplicate parameter	18	enable
		6	Invalid parameter value	19	delay_on
				20	delay_off
				30	ip
				31	port
				32	mask
		33	gateway		
		40	rtc date		
		41	rtc time		
100	Unknown error	null	null		

Tabela 15 – Resposta de Erro.

Contato

Comercial

Matriz

Porto Alegre/RS
Rua Coronel Armando Assis, 222
Brasil - CEP: 91330-010
+55 (51) 3026-2300

Filial

São Caetano do Sul/SP
Alameda Terracota, 215/1003
Brasil - CEP: 09531-190
+55 (11) 4063-4300

Assistência Técnica

assistencia.tecnica@commbox.com.br
+55 (51) 3026-2300

Suporte Técnico

suporte@commbox.com.br
+55 (51) 3026-2300