



Manual do usuário

DTN-485



ENABLING TECHNOLOGY



07300326

Khomp - Todos os direitos reservados

Índice

1. Introdução	página 3
1.1. Visão geral sobre o DTN-485	Página 3
1.2. Recursos	Página 3
1.3. Especificações	Página 4
1.4. Modo de suspensão e modo de trabalho	Página 4
1.5. LEDs e botões	Página 5
1.6. Conexões Internas	Página 6
1.7. Dimensões	Página 7
1.8. Inserindo o SIM card no DTN-485	Página 8
1.8.1. Chaves de segurança	Página 10
2. Como funciona o DTN-485	Página 12
2.1. Conexão física com os sensores	Página 12
2.2. Comunicação de dados com os sensores	Página 13
3. Configuração do DTN-485	Página 14
3.1. Configuração geral do DTN-485 via app Konfig	Página 14
3.2. Configurações Específicas para o DTN-485 via app Konfig	Página 15
4. Comandos AT específicos para o DTN-485	Página 16
4.1. Comando de debug (AT+CFGDEV)	Página 16
4.2. Comando de ação (AT+COMMANDX)	Página 17
4.3. Comando de retorno (AT+DATACUTX)	página 18
4.4. Tipos de payload (AT+PRO)	página 19
4.4.1 Formato JSON	página 20
4.4.2 Formato HEXADECIMA	página 21
4.5. Alterar o intervalo de uplink (AT+TDC)	página 22
4.6. Excluir as regras flexíveis (AT+CMDEAR)	página 22
5. Obter acesso à documentação adicional	página 23

1. Introdução

1.1. Visão geral sobre o DTN-485

O DTN-485 é um dispositivo que utiliza o protocolo RS485 para a conexão física e o protocolo MODBUS-RTU para a comunicação de dados com os sensores conectados a ele. Foi criado especialmente para aplicações IoT, onde o usuário precise enviar dados de sensores RS485/MODBUS-RTU para à rede NB-IoT.

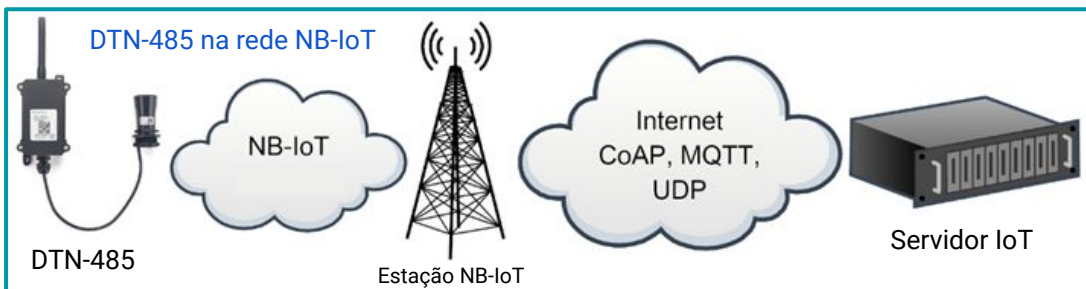
Dispõe de 15 regras flexíveis, onde são ajustadas para comunicação com os sensores conectados na sua interface RS485. Cada regra flexível equivale a um comando MODBUS-RTU que o DTN-485 enviará à um dos sensores. Atuará obrigatoriamente como o dispositivo MODBUS Master nos cenários de aplicação.

Suporta diferentes métodos de uplink, incluindo MQTT, MQTTs, UDP, TCP ou CoAP para diferentes requisitos de aplicação, e suporta uplinks para vários servidores IoT.

Conta com uma proteção IP65 (waterproof) e foi projetado para uso prolongado em situações adversas de clima.

Possui saídas ajustáveis de 3.3 V e 5 V para alimentar sensores externos, permitindo o controle dessas tensões para reduzir o consumo de energia do sistema.

É alimentado por uma bateria [Li/SOCI2](#) de 8500 mAh, assegurando vários anos de operação contínua, embora a duração da bateria seja dependente da quantidade de uplinks diários.



1.2. Recursos

- Bandas NB-IoT: B1/B2/B3/B4/B5/B8/B12/B13/B17/B18/B19/B20/B25/B28/B66/B70/B85 @H-FDD
- Conexão física através do protocolo RS485
- Comunicação de dados através do protocolo MODBUS-RTU
- Suporte à conexão direta com 1 único sensor RS485 ou à um barramento com até 30 sensores
- 15 Regras flexíveis para comunicação com diferentes sensores
- Uplink via MQTT, MQTTs, TCP ou UDP
- Configuração remota via BLE e comandos downlink
- Consumo ultra baixo de energia
- Saída controlável de tensão (3.3 V e 5 V)
- Grau de proteção IP65
- Bateria Li/SOCI2 de 8500 mAh.
- Slot para cartão Nano SIM para NB-IoT SIM

1.3. Especificações

Características comuns de DC

- Tensão de alimentação: 2,5 v até 3,6 v.
- Temperatura operacional: -40 °C até 85 °C.

Suporte para bandas NB-IoT

- B1 @H-FDD: 2100 MHz.
- B2 @H-FDD: 1900MHz
- B3 @H-FDD: 1800 MHz.
- B4 @H-FDD: 2100MHz
- B5 @H-FDD: 860MHz
- B8 @H-FDD: 900 MHz.
- B12 @H-FDD: 720MHz
- B13 @H-FDD: 740MHz
- B17 @H-FDD: 730MHz
- B18 @H-FDD: 870MHz
- B19 @H-FDD: 870MHz
- B20 @H-FDD: 800 MHz.
- B25 @H-FDD: 1900MHz
- B28 @H-FDD: 700 MHz.
- B66 @H-FDD: 2000MHz
- B70 @H-FDD: 2000MHz
- B85 @H-FDD: 700MHz

Bateria

- Bateria Li/SOCI2 não recarregável.
- Capacidade: 8500 mAh.
- Autodescarga: <1% / Ano a 25 °C.
- Corrente máxima contínua: 130 mA.
- Corrente máxima de reforço: 2 A, 1 segundo.

Consumo de energia

- Modo de suspensão: 10 µA @ 3,3 V.
- Modo de transmissão: 350 mA @ 3,3 V.

Grau de proteção

- IP65 (waterproof)

Garantias e certificações

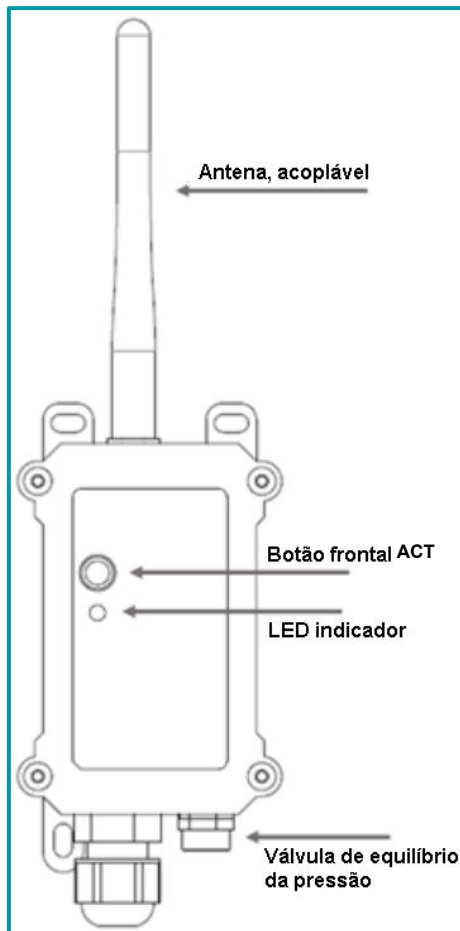
- Garantia total (legal + garantia khomp):
 - 1 ano
 - Garantia legal: 90 dias
 - Garantia Khomp: 9 meses
- Certificação Anatel
- Indústria certificada ISO 9001

1.4. Modo de suspensão e modo de trabalho

Modo de trabalho (Working Mode): Neste modo, o equipamento possui o funcionamento normal de um dispositivo NB-IoT. Ele irá ingressar na rede NB-IoT e enviará dados ao gateway. Periodicamente, entre cada amostragem, o dispositivo entrará no modo IDLE. No modo IDLE, ele terá o mesmo consumo de energia que no modo de suspensão.

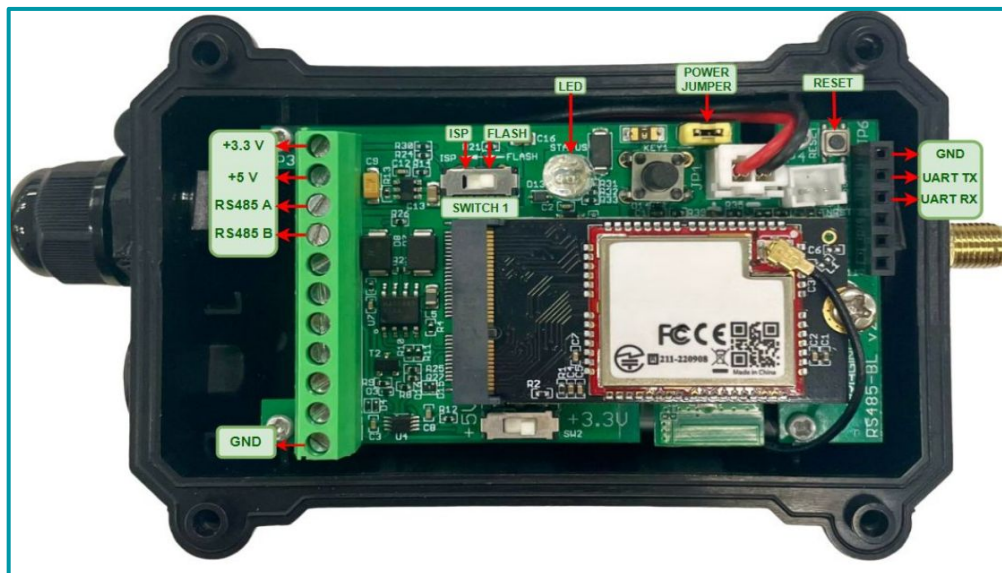
Modo de suspensão (Deep Sleep Mode): Quando o equipamento não possui conexão NB-IoT, ele entra no modo suspensão. Este modo é utilizado para economizar bateria e otimizar a vida útil do equipamento.

1.5. LEDs e botões



Ações no botão ACT	Funções	Ações
Pressionar o botão ACT de 1 segundo a 3 segundos	Enviar um uplink	Apenas se o sensor estiver conectado à rede NB-IoT, o sensor enviará um pacote de uplink e o LED irá piscar azul uma vez (esta ação leva alguns segundos para acontecer). Enquanto isso, o módulo Bluetooth estará ativo e o usuário poderá se conectar via Bluetooth para configurar o dispositivo.
Pressionar o botão ACT por mais de 3 segundos	Reiniciar Dispositivo	O LED pisca verde rapidamente 5 vezes, o dispositivo entrará no modo OTA por 3 segundos. Em seguida, ENTRA na rede NB-IoT. O LED acende verde continuamente por 5 segundos após entrar na rede. Assim que o sensor estiver ativo, o módulo Bluetooth estará ativo e o usuário poderá se conectar via Bluetooth para configurar o equipamento, independentemente de o dispositivo ingressar ou não na rede NB-IoT.
Pressionar o botão ACT rapidamente 5 vezes	Dispositivo desativado	O LED ficará aceso na cor vermelho por 5 segundos. Significa que o DTN-485 está no modo de suspensão de energia.

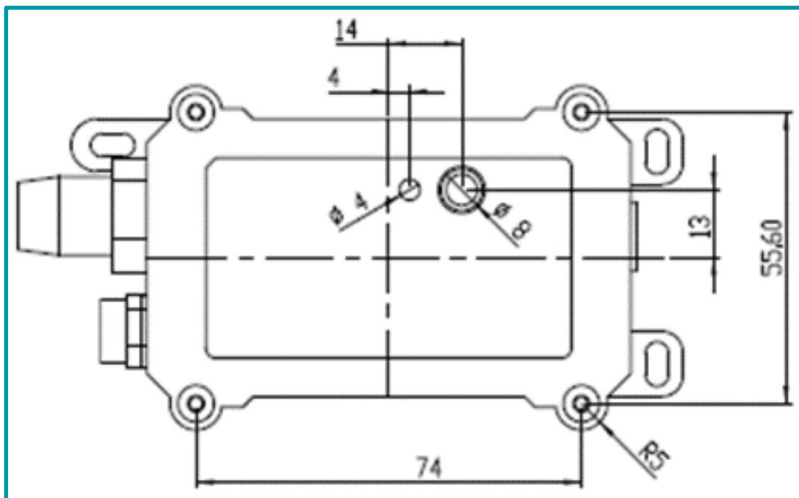
1.6. Conexões Internas



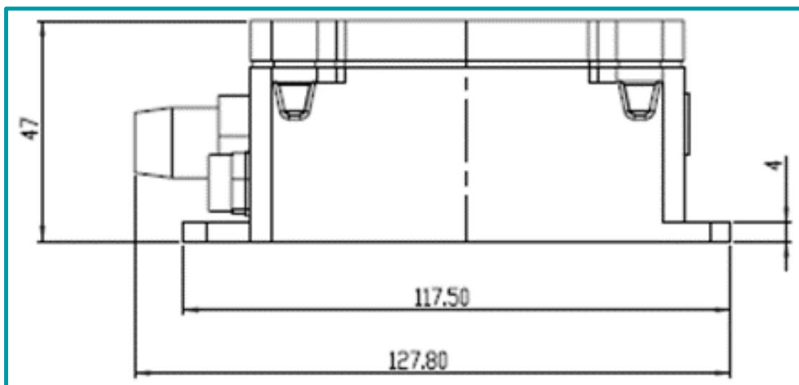
Barramento	Funções
+3.3V	Saída controlável de 3.3V DC, (nível de tensão igual ao da bateria, 2.6–3.6 V). Uso: Utilizado para alimentar sensores que requerem uma tensão de 3,3V
+5V	Saída controlável de 5V DC. Uso: Utilizado para alimentar sensores que requerem uma tensão de 5V
RS485 A RS485 B	Linhas de transmissão e recepção para o protocolo RS-485. Uso: Utilizado para conexão do dispositivo com o barramento RS485 ou diretamente ao sensor RS485
GND	GND (terra) - ponto de referência comum. Uso: Serve como ponto de referência para as tensões e caminho de retorno para a corrente elétrica.

Outras Conexões	Funções
Reset	Botão de reset. Uso: Utilizado para resetar as configurações de fábrica no dispositivo.
Power Jumper	Pinos de alimentação. Uso: Utilizado para fechar o contato nos pinos de alimentação do dispositivo
LED	LED indicador. Uso: Utilizado para verificar o status de operação do dispositivo

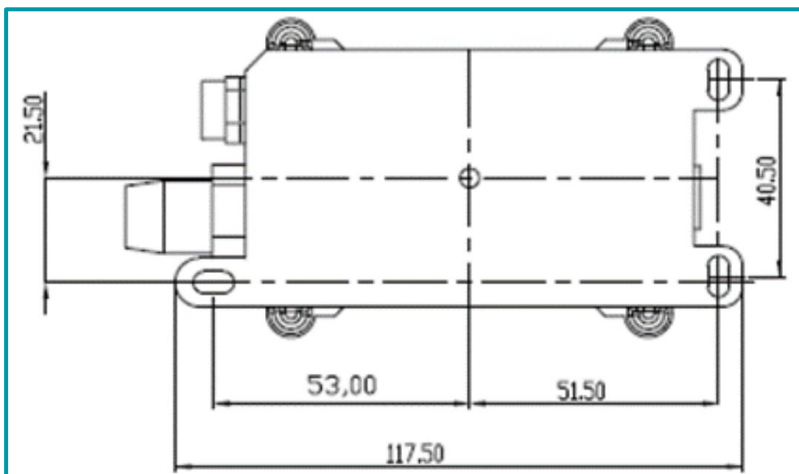
1.7. Dimensões



Legenda: Dimensões na parte frontal do DTN-485.



Legenda: Dimensões na parte lateral do DTN-485.



Legenda: Dimensões na parte traseira do DTN-485.

1.8. Inserindo o SIM card no DTN-485

No DTN-485, o SIM card é inserido na parte de baixo do módulo NB. Para acessar o local correto, é preciso remover o módulo da placa principal do dispositivo.

Um exemplo de como instalar o SIM card no dispositivo é observado a seguir:

1. Remova os 4 parafusos na parte frontal do DTN-485.



2. Remover o pino que ativa a alimentação da bateria e também o parafuso que fixa o módulo na placa principal;

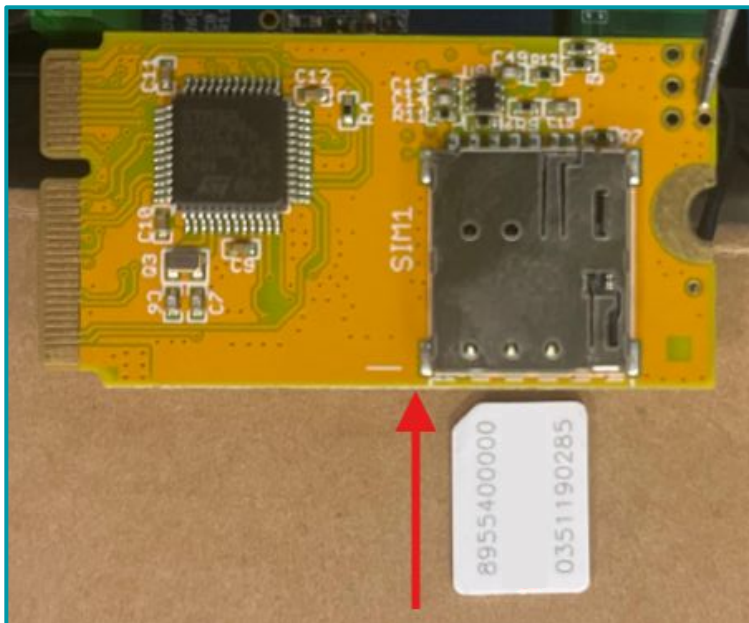


3. Insira o SIM card no slot da placa, conforme mostra a imagem a seguir.

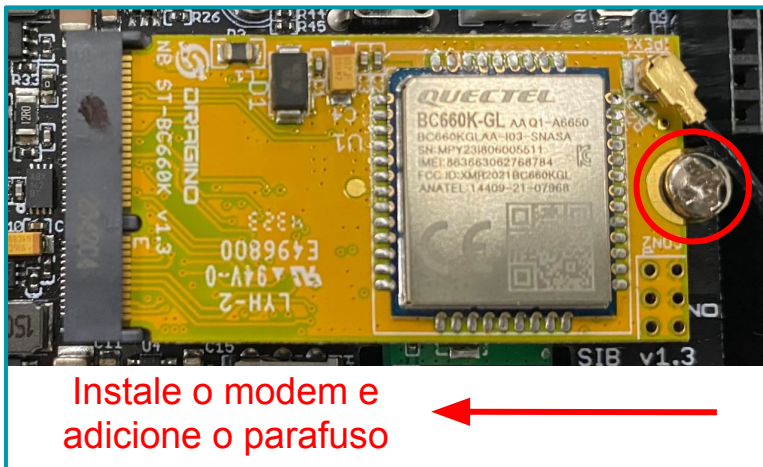


Nota

Ao inserir o SIM card corretamente, é possível escutar um "click".



4. Após aplicar todas as etapas, posicione o módulo no local de origem e fixe os parafusos novamente.



1.8.1. Chaves de segurança

Como mencionado anteriormente, o dispositivo possui um conjunto único de chaves para registro no servidor. Para ingressar o dispositivo na rede mobile, é necessário inserir as chaves no servidor e, após isso, ligar o dispositivo para que ele inicie o processo de JOIN (adesão à rede) automaticamente.

As chaves de segurança estão localizadas em uma etiqueta dentro da caixa do produto. Além das chaves, a etiqueta também contém outras chaves privadas do dispositivo, utilizadas para diferentes processos.

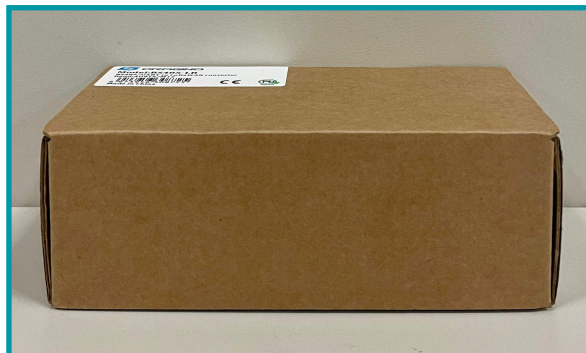


- Guarde bem as chaves de cada equipamento.
- Somente as chaves podem adicionar o endpoint na rede mobile.
- As chaves também são necessárias para alterar as configurações do dispositivo.

Abra a caixa e observa a etiqueta no lado interno da tampa (na embalagem).

Um exemplo de onde localizar a etiqueta com as chaves do DTN-485 é observado a seguir:

Caixa
fechada



Caixa
aberta



A seguir, um exemplo de onde está localizada as chaves de segurança, IMSI e Serial Number (SN):



Nota

Alguns números foram ocultados por questões de privacidade e segurança.

2. Como funciona o DTN-485

O DTN-485 funciona como um conversor, projetado para integrar sensores RS485 / MODBUS_RTU com redes NB IoT.

O sistema utiliza 2 protocolos para a integração com os sensores, um protocolo para a conexão física e o outro para a troca de dados.

2.1. Conexão física com os sensores

O DTN-485 dispõe de uma interface RS485, utilizada como meio físico para a conexão serial entre o equipamento e os sensores.

O protocolo RS485 é um padrão que define a maneira como os sinais são transmitidos em um barramento de comunicação, permitindo a conexão de múltiplos dispositivos em um único par de cabos diferencial.

A seguir, é indicada uma imagem onde mostra o DTN-485 conectado pela interface RS485. Os pinos 3 e 4 do borne interno, são disponibilizados para isso.



Nota

A ideia da imagem foi exemplificar a conexão com a interface RS485. Utilize o "passa cabos" antes de conectar os fios aos bornes da interface RS485 em um cenário real.

2.2. Comunicação de dados com os sensores

Para a comunicação de dados entre o DTN-485 e os sensores, será utilizado o protocolo MODBUS-RTU. O MODBUS-RTU (Remote Terminal Unit) é um protocolo de comunicação serial amplamente utilizado para troca de dados em sistemas de automação industrial.

Ele opera em uma configuração mestre-escravo, isso significa que um dispositivo mestre controla a comunicação e envia comandos para um ou mais dispositivos escravos. No contexto do DTN-485, o dispositivo opera como o mestre. Isso implica que o DTN-485 é responsável por iniciar e controlar a comunicação com os sensores (escravos).

O processo de comunicação é descrito a seguir:

- 1. Envio de Solicitação:** O DTN-485, atuando como mestre, envia uma solicitação MODBUS-RTU para um sensor. Essa solicitação inclui o endereço do dispositivo escravo, o código da função desejada (por exemplo, leitura de registradores), o endereço dos registradores a serem consultados e quaisquer parâmetros adicionais necessários (por exemplo, o CRC).
- 2. Recebimento de Resposta:** O sensor escravo responde com os dados solicitados ou com uma mensagem de erro, dependendo do processamento da solicitação. A resposta inclui as informações requisitadas ou um código de erro indicando problemas na solicitação.
- 3. Processamento dos Dados:** Após receber a resposta dos sensores, o DTN-485 processa as informações coletadas. Este processamento pode envolver a formatação ou agregação dos dados, conforme necessário para a transmissão.

Cada mensagem MODBUS-RTU é composta por:

- **Endereço do Dispositivo:** Identifica à qual dispositivo escravo a mensagem será entregue.
- **Código da Função:** Define a ação a ser executada, como leitura de dados ou escrita de dados.
- **Dados:** Endereço dos registradores que serão consultados no sensor.
- **CRC (Cyclic Redundancy Check):** Um código de verificação de erros que garante a integridade dos dados transmitidos.

Embora o MODBUS-RTU seja um protocolo de comunicação padronizado, a maneira como os dispositivos interpretam e utilizam esse protocolo pode variar bastante entre diferentes fabricantes e modelos, por exemplo:

- Cada sensor pode ter um mapeamento de registradores diferente, o que significa que os endereços dos registradores que armazenam os dados podem variar. Por exemplo, o registrador para leitura da temperatura em um sensor pode estar no endereço 0x1000, enquanto em outro sensor pode estar no endereço 0x2000.
- Embora os códigos de função MODBUS sejam padrão (como leitura e escrita de registradores), a forma como cada dispositivo utiliza esses códigos pode variar. Alguns dispositivos podem ter códigos de função específicos para operações particulares que não são padrão.
- Os formatos de dados (como o número de bytes utilizados para representar um valor ou o tipo de dados – inteiro, flutuante, etc.) podem variar. Por exemplo, um sensor pode enviar valores como inteiros de 16 bits, enquanto outro pode enviar valores de ponto flutuante de 32 bits.
- Cada sensor precisa ter um endereço único na rede MODBUS-RTU. O endereço é atribuído geralmente por meio de configurações físicas ou via software.

Com isso, o comando MODBUS-RTU que o DTN-485 (dispositivo mestre) deverá enviar à cada sensor, deverá ser disponibilizado pelo fabricante do sensor



Nota

Caso não encontre como foi implementado o protocolo MODBUS-RTU nas documentações do sensor, recomenda-se entrar em contato com o fabricante e solicitar essa informação.

3. Configuração do DTN-485

O DTN-485 suporta a conexão via BLE (Bluetooth) com outros dispositivos. Com isso, a Khomp disponibiliza o aplicativo **KONFIG** para realizar a configuração dos parâmetros do endpoint.

Os endpoints da linha DTN são configurados através de comandos AT. Portanto, aceitam comandos do tipo:

AT + comando = valor_do_parâmetro

Para facilitar aos usuários que possuem endpoints da linha DTN, o aplicativo Konfig possui uma série de botões predefinidos onde visam economizar tempo na configuração e deixá-la mais dinâmica e simples. O aplicativo está disponível para as plataformas Android e iOS e pode ser baixado através dos links:

- **Android:** <https://play.google.com/store/apps/details?id=com.khomp.konfig&pli=1>
- **iOS:** <https://apps.apple.com/us/app/konfig/id6739005051>

2.1. Configuração geral do DTN-485 via app Konfig

Os endpoints da linha DTN possuem a mesma base de configuração inicial. Essas configurações podem facilmente serem feitas através do aplicativo Konfig, com os botões predefinidos.

Disponibilizamos um manual a parte para este tipo de configuração, onde será encontrado a maneira correta de usar os comandos e exemplos para auxiliar no processo. A documentação para a configuração geral pode ser obtida através do endereço observado a seguir:

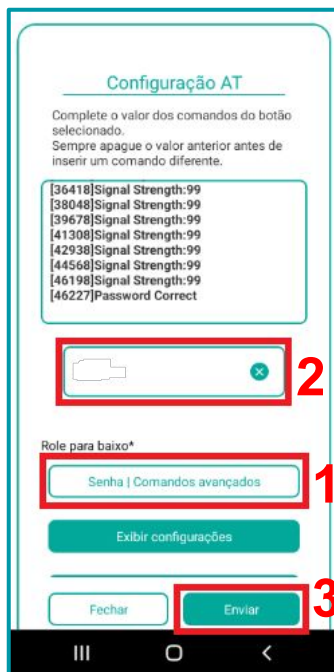
https://docs.google.com/presentation/d/1WNFs9TNmAUDxLEtKSFsrdZ6zkKZDOyjAikiBBoTqcol/edit#slide=id.g2d6c81bd0a1_1_0

3.2. Configurações Específicas para o DTN-485 via app Konfig

Como informado anteriormente, a configuração dos parâmetros nos endpoints da linha DTN é feita através de comandos AT.

O aplicativo Konfig, possui um botão onde o usuário pode informar os comandos AT de configurações específicas e também os seus valores.

A imagem a seguir possui indicações para enviar comandos AT.



Legenda:

1. **Botão Senha | Comandos Avançados:** Botão para habilitar o local de envio dos comandos específicos.
2. **Campo de input:** Local para ser inserido o comando AT.
3. **Botão Enviar:** Botão para enviar o comando AT.

Portanto, sempre que for enviado um comando específico para o DTN-300, deverão ser aplicados os procedimentos descritos a seguir:

- a. Clicar no botão "Senhas | Comandos avançados".
- b. Inserir o comando AT corretamente no "Campo de input".
- c. Clicar no botão "Enviar".

4. Comandos AT específicos para o DTN-485

4.1. Comando de debug (AT+CFGDEV)

O DTN-485 obtém os dados contendo as informações através de comandos modbus-rtu. Na maioria dos casos, os comandos são de leitura dos registradores que armazenam essas informações nos sensores. Ou seja, o DTN-485 envia a requisição de leitura do registrador e espera obter uma resposta como retorno. Essa resposta conterá o dado.

Antes de configurar no DTN-485 o comando modbus, pode-se utilizar o comando de debug para validar se a requisição enviada obtém o retorno esperado. Desta forma, o usuário consegue validar os comandos antes de ocupar a memória no dispositivo e também consegue trabalhar com a resposta do comando.

O formato do comando de debug é observado a seguir:

Tipo de comando	Formato
DEBUG	AT+CFGDEV = comando_modbus, crc_mode
Parâmetro	Descrição do parâmetro
comando_modbus	Comando modbus-rtu (em hexadecimal).
crc_mode	Indica se o DTN-485 deve calcular automaticamente o CRC-16/MODBUS do comando ou se ele será repassado junto ao comando. Pode ser: 0 ou 1, onde: 0 → Adicionado o CRC junto ao comando. 1 → CRC deve ser calculado automaticamente.

Exemplo: Supondo que o comando modbus-rtu para obter a temperatura de um sensor RS485 seja 01 03 00 00 00 01 84 0A, onde 01 03 00 00 00 01 são os bytes do comando modbus e 84 0A são os bytes do CRC deste comando. Então, para verificar se o comando obtém uma resposta correta, pode ser enviado o comando de debug. Para este caso, o comando poderia ser configurado da seguinte maneira:

Tipo de comando	Formato
DEBUG	CRC no comando: AT+CFGDEV = 010300000001840A,0 CRC automático: AT+CFGDEV = 010300000001,1

4.2. Comando de ação (AT+COMMANDX)

O comando de ação é utilizado para gravar na memória do DTN-485 as requisições modbus-rtu para obter as informações dos sensores conectados a ele. Periodicamente (à cada uplink), o DTN-485 enviará os comandos de ação gravados em sua memória aos sensores conectados a ele. Com isso, ele obterá as respostas desses sensores e formará um pacote enviado para a aplicação.

Ao todo, são 15 comandos (regras flexíveis) de ação que podem ser gravados no dispositivo e como dito anteriormente, esses comandos deverão corresponder ao comando modbus-rtu para ler os registradores que guardam as informações dos sensores.

O formato do comando de ação é observado a seguir:

Tipo de comando	Formato
AÇÃO	AT+COMMANDX = comando_modbus, crc_mode
Parâmetro	Descrição do parâmetro
X	Indica o valor da regra flexível que será gravado no equipamento. Deve ser um valor inteiro entre 1 e 15.
comando_modbus	Comando modbus-rtu (em hexadecimal).
crc_mode	Indica se o DTN-485 deve calcular automaticamente o CRC-16/MODBUS do comando ou se ele será repassado junto ao comando. Pode ser: 0 ou 1, onde: 0 → Adicionado o CRC junto ao comando. 1 → CRC deve ser calculado automaticamente.

Exemplo: Supondo que o comando modbus-rtu para obter a temperatura de um sensor RS485 seja 01 03 00 00 00 01 84 0A, onde 01 03 00 00 00 01 são os bytes do comando modbus e 84 0A são os bytes do CRC deste comando. Então, para gravar na memória do DTN-485 um comando de ação para obter a temperatura deste sensor, pode ser enviado o comando:

Tipo de comando	Formato
AÇÃO	CRC no comando: AT+COMMAND1 = 010300000001840A,0 CRC automático: AT+COMMAND1 = 010300000001,1



Nota

Note que no exemplo foi utilizado a regra flexível 1, mas poderia ser utilizado outro valor (inteiro) entre 1 e 15 para isso.

4.3. Comando de retorno (AT+DATA CUTX)


Para cada comando de ação que é gravado no DTN-485, deve ser configurado um comando de retorno. Este comando irá atuar sobre a resposta obtida pelo comando de ação. Este comando pode ser utilizado para descartar os bytes da resposta que não são interessantes para a aplicação e criar uma nova resposta, ou simplesmente, exibir toda a mensagem

O formato do comando de ação é observado a seguir:

Tipo de comando	Formato
RETORNO	AT+DATA CUTX = length_command, 2, interval_bytes
Parâmetro	Descrição do parâmetro
X	Indica o valor da regra flexível que será gravado no equipamento. Deve ser um valor inteiro entre 1 e 15. OBS: Como o comando de retorno atua sobre a resposta obtida ao comando de ação, o valor de X deve corresponder à mesma regra flexível do comando de ação configurado.
length_command	Quantidade total, em bytes, do comando de resposta obtido pelo equipamento.
2	Valor específico para que o DTN-485 interprete o comando como um comando de retorno.
interval_bytes	Intervalo de bytes que será retirado da mensagem original e irão formar a nova mensagem. Esta nova mensagem é a que será retornada ao usuário em sua aplicação.

Exemplo: Supondo que a resposta obtida pelo DTN-485 ao comando de ação seja: **01 03 02 00 F4 B9 C3**, onde **01 03 02 00 F4** é a resposta e **B9 C3** são os bytes do CRC deste comando. Supondo que os bytes do CRC não interessem para a nossa aplicação, poderíamos separar eles da resposta original e criar uma nova resposta, sem esta informação. O comando de retorno para isso seria:

Tipo de comando	Formato
RETORNO	AT+DATA CUT1 = 7,2,1~5

	Nota É possível incluir mais de um intervalo na mesma mensagem, fazendo: AT+DATA CUTX = length_command,2,interval_1, interval_2. Por exemplo: AT+DATA CUT5 = 7,2,1~3, 6~7.
--	--

4.4. Tipos de payload (AT+PRO)

Para se adequar a diferentes servidores de aplicação, o DTN-485 consegue enviar os dados em diferentes formatos e protocolos.

Os tipos de payload são divididos em 2 partes: formato e protocolo.

Formato:

- JSON
- Hexadecimal

Protocolo:

- UDP
- TCP
- MQTT

O formato do comando para definir o tipo de payload é observado a seguir:

Tipo de comando	Formato
TIPO DE PAYLOAD	AT+PRO = tipo_do_protocolo , tipo_do_formato
Parâmetro	Descrição do parâmetro
tipo do formato	Indica o tipo do formato da mensagem que será enviada. Os tipos podem ser: <ul style="list-style-type: none">• JSON → Tipo 5• HEXADECIMAL → Tipo 0
tipo do protocolo	Indica o tipo do protocolo utilizado pelo equipamento para envio da mensagem. Os tipos podem ser: <ul style="list-style-type: none">• UDP → Tipo 2• MQTT → Tipo 3• TCP → Tipo 4

Exemplo 1: Supondo que a o usuário deseja configurar o DTN-485 para enviar os dados para um servidor de aplicação **MQTT** no formato **JSON**, o comando de payload para isso seria:

Tipo de comando	Formato
TIPO DE PAYLOAD	AT+PRO = 3,5

Exemplo 2: Supondo que a o usuário deseja configurar o DTN-485 para enviar os dados para um servidor de aplicação **TCP** no formato **HEXADECIMAL**, o comando de payload para isso seria:

Tipo de comando	Formato
TIPO DE PAYLOAD	AT+PRO = 4,0

4.4.1 Formato JSON

Para o formato JSON, o DTN-485 irá enviar uma mensagem ao servidor de aplicação da seguinte forma:

```
{"IMEI":"imei_addr","Model":"device_model","Payload":"data_payload","battery":battery_level,"signal":network_signal}
```

A explicação de cada parâmetro é observado a seguir:

Parâmetro	Descrição do parâmetro
imei_addr	Endereço de IMEI do dispositivo. O IMEI é o identificador do equipamento, portanto, ele é único para cada dispositivo.
device_model	Indica o modelo do dispositivo. Para caso do DTN-485 será: RS485-NB.
data_payload	Para o caso do DTN-485 serão os dados obtidos nas leituras dos registrados dos dispositivos conectados ao DTN-485. Esses dados são obtidos configurando os comandos AT+COMMANDX e AT+DATA CUTX.
battery_level	Indicará o nível de bateria do dispositivo.
network_signal	Indicará a qualidade do sinal obtido na última mensagem.

Exemplo: Um exemplo do comando JSON pode ser observado abaixo:

```
Value [icon] ^
<> [menu] QoS: 0
13/12/2024 12:18:57
{
  "IMEI": "863663062782900",
  "Model": "RS485-NB",
  "Payload": "0101030200ebf80b",
  "battery": 3.388,
  "signal": 22
}
```

Neste caso, o DTN-485 foi configurado para obter a temperatura de um sensor que está conectado em sua interface RS485.

O comando modbus de resposta pode ser observado em Payload: **01030200ebf80b**.

É possível observar a intensidade do sinal e também o nível de bateria atual do dispositivo.

4.4.2 Formato HEXADECIMAL

Para o formato HEXADECIMAL, o DTN-485 irá enviar uma mensagem ao servidor de aplicação da seguinte forma:

imei_addr device_version battery_level network_signal gpio_exit_level gpio_exit_flag timestamp payload_version data_payload

A explicação de cada parâmetro é observado a seguir:

Parâmetro	Tamanho	Descrição do parâmetro
imei_addr	8 bytes	Endereço de IMEI do dispositivo. O IMEI é o identificador do equipamento, portanto, ele é único para cada dispositivo.
device_version	2 bytes	Inclui a versão de hardware e a versão de firmware. O "Higher byte" indicará o modelo do sensor e o "Lower byte" indicará a versão de software.
battery_level	2 bytes	Indicará o nível de bateria do dispositivo.
network_signal	1 byte	Indicará a "qualidade" do sinal NB-IoT, onde: 0 → -113dBm ou menor; 1 → -111 dBm; 2 à 30 → 109 dBm à -53dBm; 31 → -51 dBm 99 → Sem sinal ou "não detectado"
GPIO_EXIT level	1 byte	Indica se o GPIO_EXTI é utilizado como um pino de interrupção. Não se aplica ao DTN-485.
GPIO_EXIT flag	1 byte	Indica se a mensagem gerada é por consequência do pino de interrupção ou não. Não se aplica ao DTN-485.
Timestamp	4 bytes	Indica o horário em que o dispositivo enviou a mensagem.
payload_version	1 byte	Indica como o dispositivo irá tratar diferentes decodes (se forem inseridos no dispositivo).
data_version	N bytes	Para o caso do DTN-485 serão os dados obtidos nas leituras dos registrados dos dispositivos conectados ao DTN-485. Esses dados são obtidos configurando os comandos AT+COMMANDX e AT+DATACUTX.

Exemplo: Um exemplo do comando HEXADECIMAL pode ser observado na imagem a seguir:



4.5. Alterar o intervalo de uplink (AT+TDC)

Este comando permite alterar o intervalo em que o dispositivo enviará as mensagens ao servidor de aplicação.



Nota

Por padrão, o intervalo de uplink do DTN vem configurado para 2 horas.

O formato do comando para definir o tipo de payload é observado a seguir:

Tipo de comando	Formato
INTERVALO DE UPLINK	AT+TDC = intervalo (em segundos)
Parâmetro	Descrição do parâmetro
intervalo	Define o intervalo (em segundos) em que o dispositivo irá enviar o uplink de dados.

Exemplo 1: Supondo que a o usuário deseja configurar o DTN-485 para enviar os dados para um servidor de aplicação à cada 20 minutos, o comando de payload para isso seria:

Tipo de comando	Formato
INTERVALO DE UPLINK	AT+TDC = 1200

4.6. Excluir as regras flexíveis (AT+CMDEAR)

Este comando permite excluir as regras flexíveis configuradas no dispositivo.
O formato do comando para definir o tipo de payload é observado a seguir:

Tipo de comando	Formato
EXCLUIR REGRAS FLEXÍVEIS	AT+CMDEAR= flexible_rule_interval
Parâmetro	Descrição do parâmetro
flexible_rule_interval	Define o intervalo das regras flexíveis que serão excluídas.

Exemplo 1: Supondo que a o usuário deseja excluir do DTN-485 as regras flexíveis de 2 a 5, o comando de payload para isso seria:

Tipo de comando	Formato
EXCLUIR REGRAS FLEXÍVEIS	AT+CMDEAR = 2,5

5. Obter acesso à documentação adicional

Você encontra o manual e outros documentos em nosso site, www.khomp.com. Veja a seguir como se cadastrar e acessar nossa documentação:

Para usuários que não possuem cadastro:

1. No site da Khomp, acesse o menu "Suporte Técnico" → "Área restrita".
2. Clique em "Inscreva-se".
3. Escolha o perfil que melhor o descreve.
4. Cadastre seu endereço de e-mail. É necessário utilizar um e-mail corporativo.
5. Preencha o formulário que será enviado ao seu e-mail. Caso não tenha recebido em sua caixa de entrada, confira sua caixa de spam.
6. Siga os passos descritos a seguir para fazer login na área restrita.

Para usuários que possuem cadastro:

1. Acesse o menu "Suporte Técnico" → "Área restrita".
2. Faça login com seu endereço de e-mail e senha cadastrada.
3. Acesse a opção Documentos. Você será direcionado à Wiki da Khomp.

Você também pode entrar em contato com nosso suporte técnico através do e-mail suporte.iot@khomp.com, pelo telefone +55 (48) 37222930 ou WhatsApp +55 (48) 999825358.

"Incorpora produto homologado pela Anatel sob número 07517-22-03237"

- Este equipamento não tem direito a proteção contra interferência prejudicial e não pode causar interferências em sistemas devidamente autorizados.
- Este equipamento não é apropriado para uso em ambientes domésticos, pois poderá causar interferências eletromagnéticas que obrigam o usuário a tomar medidas para minimizar estas interferências.

Para informações do produto homologado, acesse o site: <https://sistemas.anatel.gov.br/sch>



Rua Joe Collaço, 253 - Florianópolis, SC
+55 (48) 3722.2930
+55 (48) 999825358 **WhatsApp**
suporte.iot@khomp.com