

Protocolo de comunicação

GDC01 – Gerenciados de duchas



Protocolo de
comunicação

DT 754



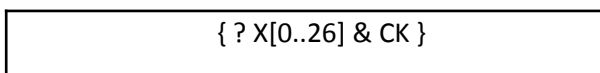
| | |
|-----------------------------------|-----------|
| 1 Introdução | 3 |
| 2 Estrutura do comando | 3 |
| 3 Comandos | 4 |
| 3.1 Bloquear acionamento | 4 |
| 3.2 Liberar acionamento | 5 |
| 3.3 Limpar memória do equipamento | 6 |
| 3.4 Gravação de senha | 7 |
| 3.5 Gravação de cartão RFID | 8 |
| 3.6 Status | 9 |
| 4 Cálculo de Checksum | 11 |

1 Introdução

- Todos os comandos são iniciados com o caractere "{" e finalizados com o caractere "}";
- Após o caractere de início, o próximo é o tipo de comando que informa qual função o equipamento GDC01 deverá executar;
- A comunicação com o GDC é inteiramente independente e somente possível via comunicação ethernet no protocolo TCP/IP utilizando a porta 888 e inicialmente no ip 192.168.0.95.

2 Estrutura do comando

Todos os comandos são definidos como a figura abaixo, eles consistem em limitadores, cabeçalho, parâmetros e um checksum, respectivamente.



- { : Inicializador do comando;
- ? : Caractere único que indica o tipo de comando:
 - B : bloquear acionamento(s);
 - L : liberar acionamento(s);
 - R : limpar memória do equipamento;
 - S : gravação de senhas;
 - C : gravação de cartões RFID;
 - X : consulta de status dos acionamentos;
 - Z : consulta de status do equipamento.
- X[0..26] : Dados do comando;
- & : Indicador de finalização dos dados do comandos;
- CK : Os caracteres delimitadores "{" e "}" são excluídos do cálculo e os caracteres restantes são somados, mas para a soma é necessário usar seu valor em hexadecimal. O checksum é calculado com base 256;
- } : Finalizador do comando.

Nas estruturas de cada um dos comandos listados a seguir, no campo de estrutura do comando e da resposta, os caracteres em **negrito** são sempre fixos e os demais variam de acordo com a estrutura de cada comando. A grande parte dos campos são representados como caracteres, entretanto os campos com o prefixo "0x" devem ser preenchidos com valores hexadecimais, em cada comando será descrita uma forma com caracteres e valores hexadecimais e uma outra forma somente com os valores hexadecimais.

3 Comandos

3.1 Bloquear acionamento

Este comando é utilizado para bloquear todos os acionamentos ou cada um deles individualmente. Ao bloquear um acionamento, mesmo que o usuário passe um cartão ou insira uma senha válidos, a ducha não será acionada. Este comando deve ser utilizado quando a ducha se encontra indisponível, seja por manutenção ou mau funcionamento.

- **Cabeçalho:** B1
- **Estrutura do comando:**
 {B1N&CK}
 - B1 [02]: Cabeçalho do comando.....(caractere)
 - N [01]: Número do acionamento (0 a 8, onde 0 = todos).....(decimal)
 - & [01]: Caractere de fim de dados.....(caractere)
 - CK [02]: Checksum.....(hexadecimal)
- **Resposta de bloqueio realizado com sucesso:**
 RX: { B N T & 0xCK }
- **Resposta acionamento já bloqueado:**
 RX: { B N F & 0xCK }
- **Exemplos:**
 - TX: {B13&CC} = Bloquear o terceiro acionamento
 - RX: {B3T&EF} = Acionamento bloqueado com sucesso

 - TX: {B10&C9} = Bloquear todos os acionamentos
 - RX: {B0T&EC} = Todos os acionamentos bloqueados com sucesso

- **Mesmos exemplos enviados em hexadecimal:**

TX: 0x7B 0x42 0x31 0x33 0x26 0x43 0x43 0x7D = Bloquear o terceiro acionamento

RX: {B3T&EF} = Acionamento bloqueado com sucesso

TX: 0x7B 0x42 0x31 0x30 0x26 0x43 0x39 0x7D = Bloquear todos os acionamentos

RX: {B0T&EC} = Todos os acionamentos bloqueados com sucesso

3.2 Liberar acionamento

Este comando é utilizado para desfazer o bloqueio de um acionamento, ou seja, realizar a liberação do mesmo. É possível realizar a liberação individualmente ou de todos no mesmo comando. Após a liberação de determinado acionamento, a ducha conectada ao mesmo voltar a estar disponível para a utilização mediante inserção de senha ou cartão RFID.

- **Cabeçalho:** L1

- **Estrutura do comando:**

{L1N&CK}

- L1 [02]: Cabeçalho do comando.....(caractere)
- N [01]: Número do acionamento (0 a 8, onde 0 = todos).....(decimal)
- & [01]: Caractere de fim de dados.....(caractere)
- CK [02]: Checksum.....(hexadecimal)

- **Resposta de liberação realizada com sucesso:**

RX: { L N T & 0xCK }

- **Resposta acionamento já liberado:**

RX: { L N F & 0xCK }

- **Exemplos:**

TX: {L13&D6} = Liberar o terceiro acionamento

RX: {L3T&F9} = Acionamento liberado com sucesso

TX: {L10&D3} = Liberar todos os acionamentos

RX: {L0T&F6} = Todos os acionamentos liberados com sucesso

- **Mesmos exemplos enviados em hexadecimal:**

TX: 0x7B 0x4C 0x31 0x33 0x26 0x44 0x46 0x7D = Liberar o terceiro acionamento

RX: {B3T&EF} = Acionamento liberado com sucesso

TX: 0x7B 0x4C 0x31 0x30 0x26 0x44 0x33 0x7D = Liberar todos os acionamentos

RX: {B0T&EC} = Todos os acionamentos liberados com sucesso

3.3 Limpar memória do equipamento

Este comando é utilizado para limpar a memória do equipamento. Todas as informações salvas no equipamento serão excluídas tais como configurações, cartões e senhas.



Atenção: Não há possibilidade de recuperação dessas informações.

- **Cabeçalho:** R
- **Estrutura do comando:**
 - {R&CK}
 - R [01]: Cabeçalho do comando.....(caractere)
 - & [01]: Caractere de fim de dados.....(caractere)
 - CK [02]: Checksum.....(hexadecimal)
- **Resposta de limpeza realizada com sucesso:**

RX: { R T & 0xCC }
- **Resposta de erro na limpeza, pode ser que tenha algum bloco danificado:**

RX: { R T & 0xBE }
- **Exemplos:**

TX: {R&78} = Limpar memória do equipamento

RX: {RT&CC} = Memória limpa com sucesso

TX: {R&78} = Limpar memória do equipamento

RX: {RF&BE} = Memória não pode ser limpa
- **Mesmo exemplo enviado em hexadecimal:**

TX: 0x7B 0x52 0x26 0x37 0x38 0x7D = Limpar memória do equipamento

RX: {RT&CC} = Memória limpa com sucesso

3.4 Gravação de senha

Este comando é utilizado para gravar uma senha no equipamento, a qual posteriormente será utilizada pelo usuário para solicitar o início de um banho. Ao gravar senha, deverão ser informados também o tempo de acionamento, ducha desejada e quantas vezes a senha será válida.

- **Cabeçalho: S**
- **Estrutura do comando:**
{S 0x0F 0xEE 0x00 0x00 0xN1 0xN2 0xN3 0xN4 0xN5 0xN6 0xDE 0xDE 0xMM 0xSS 0xND 0xQQ 0x0F & 0xCK}
 - S [01]: Cabeçalho do comando.....(caractere)
 - 0x0F [02]: Campo fixo.....(hexadecimal)
 - 0xEE [02]: Campo fixo.....(hexadecimal)
 - 0x00 [02]: Campo fixo.....(hexadecimal)
 - 0x00 [02]: Campo fixo.....(hexadecimal)
 - N1..N6 [12]: Senha.....(hexadecimal)
 - 0xDE [02]: Campo fixo.....(hexadecimal)
 - 0xDE [02]: Campo fixo.....(hexadecimal)
 - 0xMM [02]: Duração em minutos.....(hexadecimal)
 - 0xSS [02]: Duração em segundos.....(hexadecimal)
 - 0xND [02]: Número da ducha a ser acionada.....(hexadecimal)
 - 0xQQ [02]: Quantidade de vezes de validade da senha.....(hexadecimal)
 - 0x0F [02]: Campo fixo.....(hexadecimal)
 - & [01]: Caractere de fim de dados.....(caractere)
 - CK [02]: Checksum.....(hexadecimal)
- **Resposta de senha gravada com sucesso:**
RX: { S T & 0xCD }

- **Resposta de senha já gravada:**

RX: { S N & 0xC7 }

- **Exemplos:**

TX: { S 0x0F 0xEE 0x00 0x00 0x04 0x08 0x08 0x05 0x00 0x01 0xDE 0xDE 0x05 0x00 0x05 0x01 0x0F & 0x66 } = Gravar a senha 488501 com duração de 5 minutos e 0 segundos, ducha 05 e com validade de 1 vez

RX: {ST&CD} = Senha gravada com sucesso

- **Mesmo exemplo enviado em hexadecimal:**

TX: 0x7B 0x53 0x0F 0xEE 0x00 0x00 0x04 0x08 0x08 0x05 0x00 0x01 0xDE 0xDE 0x05 0x00 0x05 0x01 0x0F 0x26 0x36 0x36 0x7D = Gravar a senha 488501 com duração de 5 minutos e 0 segundos, ducha 05 e com validade de 1 vez

RX: {ST&CD} = Senha gravada com sucesso

3.5 Gravação de cartão RFID

Este comando é utilizado para gravar um cartão RFID no equipamento, a qual posteriormente será utilizado pelo usuário para solicitar o início de um banho. Ao gravar um cartão, deverão ser informados também o tempo de acionamento, ducha desejada e quantas vezes o cartão será válido.

- **Cabeçalho:** C

- **Estrutura do comando:**

{C 0x43 0x0F 0xEE 0xN1 0xN2 0xN3 0xN4 0xN5 0xN6 0xN7 0xN8 0xDE 0xDE 0xMM 0xSS 0xND 0xQQ 0x0F & 0xCK}

- C [01]: Cabeçalho do comando.....(caractere)
- 0x43 [02]: Campo fixo.....(hexadecimal)
- 0x0F [02]: Campo fixo.....(hexadecimal)
- 0xEE [02]: Campo fixo.....(hexadecimal)
- N1..N8 [12]: Tag RFID.....(hexadecimal)
- 0xDE [02]: Campo fixo.....(hexadecimal)

- 0xDE [02]: Campo fixo.....(hexadecimal)
- 0xMM [02]: Duração em minutos.....(hexadecimal)
- 0xSS [02]: Duração em segundos.....(hexadecimal)
- 0xND [02]: Número da ducha a ser acionada.....(hexadecimal)
- 0xQQ [02]: Quantidade de vezes de validade da senha.....(hexadecimal)
- 0x0F [02]: Campo fixo.....(hexadecimal)
- & [01]: Caractere de fim de dados.....(caractere)
- CK [02]: Checksum.....(hexadecimal)
- **Resposta de cartão gravado com sucesso:**
RX: { C T & 0xBD }
- **Resposta de cartão já gravado:**
RX: { C N & 0xB7 }
- **Exemplos:**
TX: { C 0x43 0x0F 0xEE 0xB3 0x28 0x09 0x09 0xF0 0xD4 0x40 0x93 0xDE 0xDE 0x05 0x00 0x05 0x01 0x0F & 0xC0 } = Gravar o cartão B3280909F0D44093 com duração de 5 minutos e 0 segundos, ducha 05 e com validade de 1 vez
RX: {CT&BD} = Cartão gravado com sucesso
- **Mesmo exemplo enviado em hexadecimal:**
TX: 0x7B 0x43 0x0F 0xEE 0xB3 0x28 0x09 0x09 0xF0 0xD4 0x40 0x93 0xDE 0xDE 0x05 0x00 0x05 0x01 0x0F 0x26 0x43 0x30 0x7D = Gravar o cartão B3280909F0D44093 com duração de 5 minutos e 0 segundos, ducha 05 e com validade de 1 vez
RX: {CT&BD} = Cartão gravado com sucesso

3.6 Status

Este comando é utilizado para pegar o status de cada ducha, ou seja, se ela está bloqueada, em funcionamento, acionada e etc.

- **Cabeçalho: X**

- **Estrutura do comando:**

{X & 0xCK}

- X [01]: Cabeçalho do comando.....(caractere)
- & [01]: Caractere de fim de dados.....(caractere)
- CK [02]: Checksum.....(hexadecimal)

- **Resposta:**

Este comando retorna uma estrutura para cada ducha, onde será representada por:
[A, B, C, D]

Onde:

- A [01]: Status da ducha.....(caractere)
 - 0 = ducha disponível para uso
 - 1 = ducha acionada
 - 2 = ducha bloqueada
 - 3 = ducha em tempo de espera, ou seja, esperando acionamento
- B [01]: Tempo faltante ou restante.....(inteiro)
 - se primeiro parâmetro (A) estiver 0 ou 2, B será zerado
 - se primeiro parâmetro (A) estiver 1, B é o tempo que falta para acabar o banho (milisegundos)
 - se primeiro parâmetro (A) estiver 3, B é o tempo que falta para o acionamento (milisegundos)
- C [01]: Vezes que a senha/cartão ainda está válido.....(hexadecimal)
 - se primeiro parâmetro (A) estiver 0 ou 2, C será zerado
 - se primeiro parâmetro (A) estiver 1 ou 3, C será a quantidade de vezes que a senha ou cartão estão disponíveis
- D [01]: Senha ou cartão que liberou a ducha.....(hexadecimal)
 - se primeiro parâmetro (A) estiver 0 ou 2, D será zerado
 - se primeiro parâmetro (A) estiver 1 ou 3, D será a senha ou cartão que liberou a ducha

- **Exemplo:**

TX: {X&7E}

RX:

{X[1,100,2,123456][1,500,1,B3280909BC346582][2,0,0,0][2,0,0,0][3,50,0,123456][0,0,0,0][0,0,0,0][0,0,0,0]&B1}

4 Cálculo de Checksum

Somatório em hexadecimais dispostos em dois caracteres adicionados às linhas de comando e resposta do equipamento, esse somatório garante que os dados estejam íntegros ao serem recebidos nas extremidades de comunicação (PC ↔ GDC).

O cálculo do checksum é realizado com um acumulador, onde são acrescidos os valores ASCII de cada caractere presente após o caractere de início "{" e de final "}" de bloco nas linhas de comando e resposta. Campos com mais de dois dígitos hexadecimais devem ser tomados de dois a dois.

Para manter um tamanho fixo de dois caracteres, as casas mais significativas são removidas do resultado do somatório, até que seu tamanho preestabelecido seja respeitado, ou seja, sempre é necessário pegar os dois caracteres mais à direita, sendo então nosso checksum calculado com base 256.

- **Exemplo de um comando de gravação de senha:**

TX: {S 0x0F 0xEE 0x00 0x00 0x04 0x08 0x08 0x05 0x00 0x01 0xDE 0xDE 0x05 0x00 0x05 0x01 0x0F & 0x66 }

53 + 0F + EE + 00 + 00 + 04 + 08 + 08 + 05 + 00 + 01 + DE + DE + 05 + 00 + 05 + 01 + 0F + 26 = 366

Depois de feito o cálculo é necessário colocar o checksum e depois os caracteres de início e fim do comando, então o comando fica da maneira que está representado no campo TX acima.

Em alguns cenários o mais fácil é enviar todo o comando em hexadecimal, até os caracteres de início "{", fim "}", cabeçalho "S" e de final do campo de dados "&". Abaixo será descrito o comando todo em hexadecimal, o cálculo de checksum segue da mesma maneira, o que mudará será no momento de colocar o valor do checksum no comando.

A soma dos valores hexadecimais do exemplo é 366, então o checksum é 66. Quando enviado o comando com os caracteres, o checksum será enviado 0x66 mesmo, mas quando



todo o comando está em hexadecimal é necessário pegar o valor 66 e convertê-lo em hexadecimal então o valor ficará 0x36 0x36, porque 6 em hexadecimal é 0x36.

- **Exemplo de um comando de gravação de senha:**

TX: 0x7B 0x53 0x0F 0xEE 0x00 0x00 0x04 0x08 0x08 0x05 0x00 0x01 0xDE 0xDE 0x05
0x00 0x05 0x01 0x0F 0x26 0x36 0x36 0x7D

Onde 0x7B = "{" e 0x7D = "}" e o checksum ficou 0x36 0x36.



Protocolo de Comunicação
Gerenciador de Duchas Companytec

DT 754
Revisão: 03
23/04/2025



Companytec Automação e Controle Ltda.
Av. Ferreira Viana, 1421 - Areal - 96080-000 - Pelotas - RS

www.companytec.com.br

Fone: (53) 3284-8129

desenvolvimento@companytec.com.br