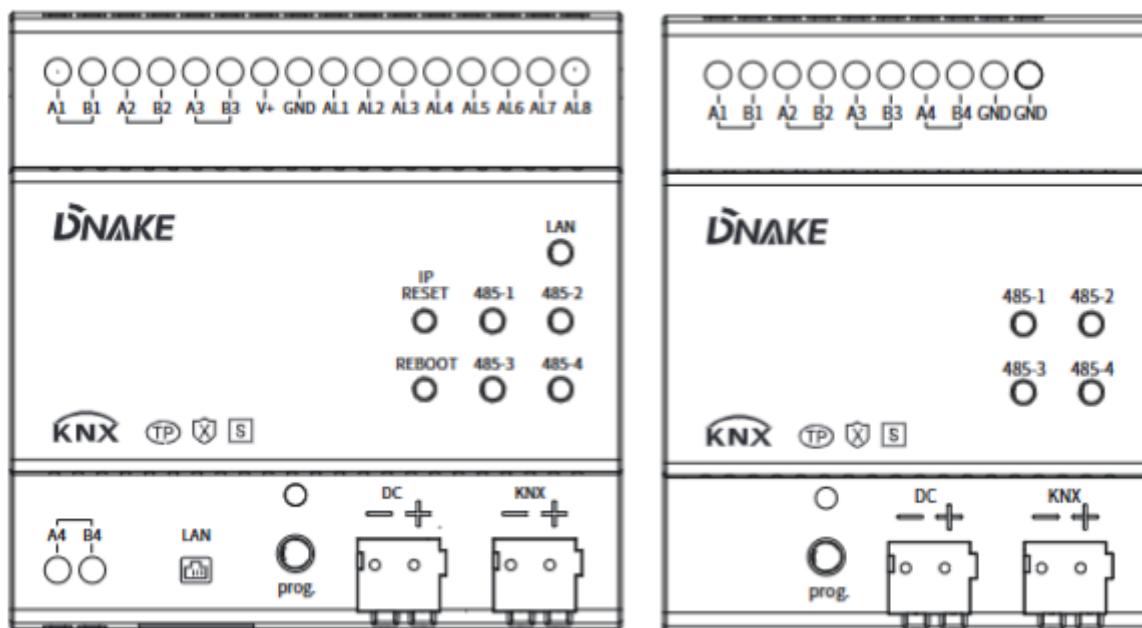


KNX-485 转化模块数据库

配置说明书

SH-KCR-DAC001



目录

- 1 恢复出厂设置 1
- 2 启用数据安全 1
- 3 通用设置 3
- 4 通道配置 3
 - 4.1 通道基本设置 3
 - 4.2 节点功能配置 4
 - 4.2.1 报文方向、读模式、透传模式、延时定时 5
 - 4.2.2 报文内容、匹配透传方式、数据类型 DPT 5
- 5 配置示例 12
 - 例 1: 某 485 设备开关机 485 12
 - 例 2: 某 485 设备控制新风挡位 13
 - 例 3: 某 485 调光设备设置亮度和读状态 14
 - 例 4: 某 485 温控设备设置温度和读状态 16

1 恢复出厂设置

可通过长按两次编程键使设备恢复出厂设置（清除数据库配置信息）。第一次长按直至编程指示灯闪烁，松开，再次长按直至指示灯常亮，松开后设备恢复出厂设置并重启，重启时指示灯闪烁。

2 启用数据安全

数据库支持开启安全调试。
若需要启用/禁用，选择设备数据库-属性-配置-安全调试-激活/禁用。

配置

备注

信息

名称

1路通用调光执行器

物理地址

描述

最后修改

2026-01-04 10:34

最后下载的

-

系列号

-

安全调试

禁用

添加设备许可证书

安全调试

禁用

激活

禁用

第一次启用（激活）安全时，会弹出提示设定项目（工程）密码，这个密码是任意自定的，请牢记。若工程中无设备启用安全，则无需密码；若至少有一个，则必须设置密码。

设定项目密码 KNX485转化模块

新密码

弱

重复密码

密码匹配

为确保通讯安全，您必须提供项目的密码，才能使项目中存储的密钥得到保护。点击“取消”按钮来取消在此项目的IP骨干网使用安全保护。

好的密码应包括

八个或更多字符

一个或多个数字

一个或多个大写字符

一个或多个小写字符

一个或多个特殊字符

厌倦了每次都要输入项目密码试试我们的项目密码管理器吧

了解更多

设置密码

取消

1 / 18

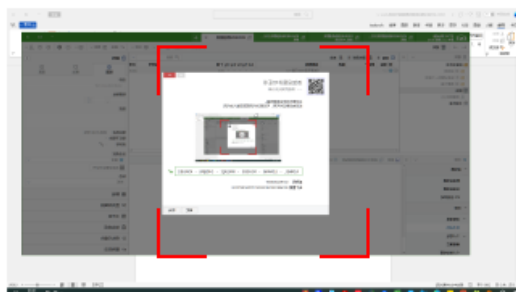
设置好工程密码后, 还需对每个设备对应的数据库添加许可证  添加设备许可证。每个设备许可证唯一, 可在设备的标签贴纸二维码上获取。扫码或手动输入。



添加设备许可证

--- 1路通用调光执行器

此设备支持安全数据传输。
如您有设备证书可用, 可扫描QR代码或现在输入该代码。



AIMWEL - LDN4AB - MCA5B5 - NRGUQC - C4DQMG - KXA5B2

系列号 0219:622D636F

出厂密钥 0016081D0F5B13520217070619570743

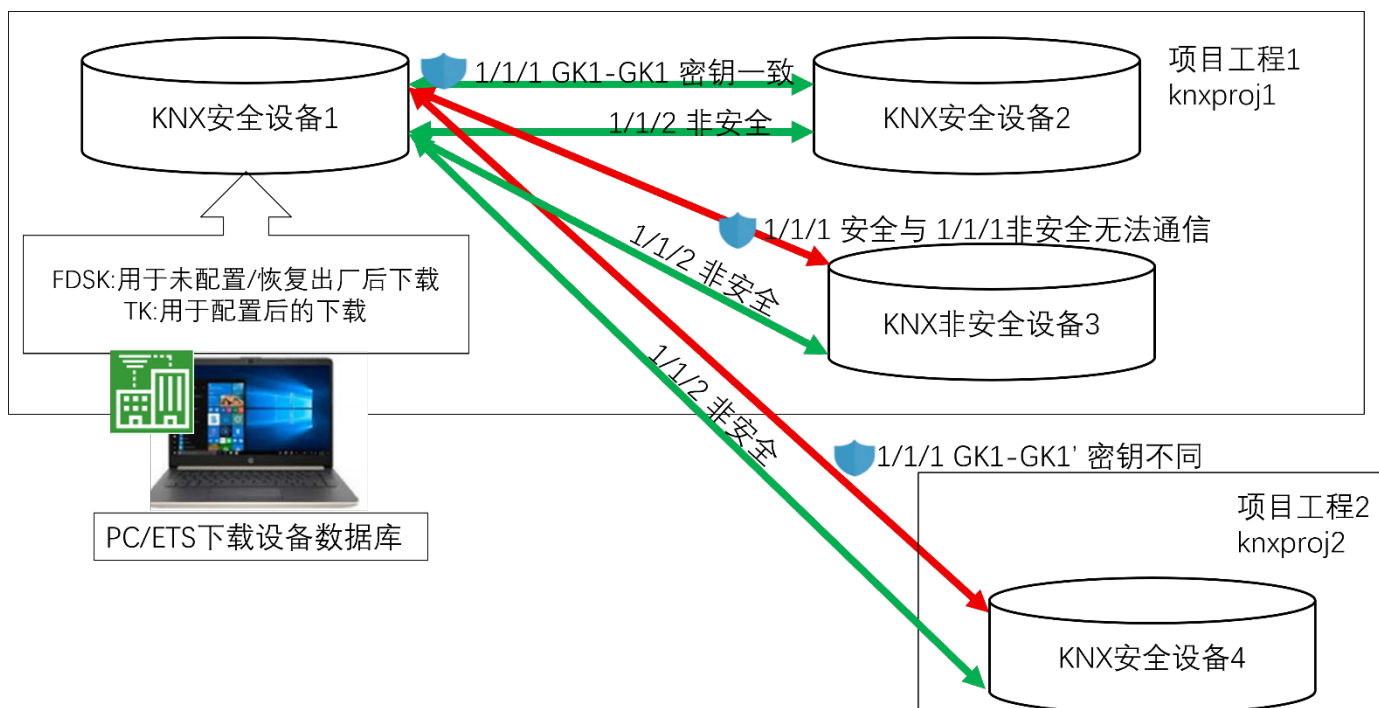
确定

取消

注: 许可证仅在设备未配置时使用, 下载后许可证不再生效 (FDSK 替换为 TK), 若更换物理地址再下载或源工程丢失, 则需将设备恢复出厂设置才能再次下载。

安全模式下的设备是闭环的, 同一群组地址安全与非安全的无法进行通信。

安全系统下设备功能控制和反馈采用群组地址密钥(Group Key, GK), 下载数据库修改设备配置使用出厂密钥(Factory Default Set key, FDSK)或工具密钥 (Tool Key, TK)。群组密钥和工具密钥由 ETS 根据不同工程分配随机值, 故不同工程间安全设备的群组也无法进行通信。



3 通用设置

1.4.1 KNX-485 转换模块 > 通用设置

通用设置

+ 通道1

+ 通道2

+ 通道3

+ 通道4

通道1

通道2

通道3

通道4

心跳周期(1~240s,0:无效)

上电启动时延(2-200s)

✓

✓

✓

✓

0

2

秒(s)

秒(s)

- 参数说明
- (1)通道启用/使能: 4 个通道独立启用/禁用, 对应设备上 4 组 485+和 485-接线口
- (2)心跳周期: 设备正常工作时定期往总线上报状态值“1”。
心跳间隔周期: 范围 1~240s。0 表示禁用。
- (3)上电启动时延: 设备上电后到开始工作等待时间, 范围 2~240s。

4 通道配置

4.1 通道基本设置

1.4.1 KNX-485 转换模块 > 通道1

通用设置

+ 通道1

+ 通道2

+ 通道3

+ 通道4

功能

描述/备注

波特率

数据位(长度)

停止位(长度)

奇偶校验

节点/对象数量

帧校验(CRC)

485帧发送间隔

KNX <-> RS485

19200

8

1

无

2

无

禁用

bits/s

- 参数说明
- (1)描述/备注: 可注释配置此通道的一些信息
- (2)波特率: 可选 1200/2400/4800/9600/19200/38400/56000/57600/115200。默认 19200bps。
- (3)数据位: 可选 7bit/8bit/9bit。默认 8bit。
- (4)停止位: 可选 0.5bit/1bit/1.5bit/2bit。默认 1bit。
- (5)奇偶校验:
可选无校验/偶校验/奇校验。默认无校验。若数据位选择 9bit, 只能选择无校验。若数据位选择 7bit, 则 HEX 数值范围在 0x0~0x7F(127)。
- (6)节点/对象数量:
485 节点/KNX 组对象数量。范围: 0~50, 每个通道最多 50 个, 4 个通道最多共 200 个。每个节点可视为一个 KNX 组对象匹配一条或一类或多条的 485/KNX 转化报文。默认 2。

(7) 帧校验:

485 整帧报文的校验。可选无校验、XOR (异或)、XOR-FF (异或再异或 FF)、SUM (和, 取低字节)、CRC-16/MODBUS、CRC-16/CCITT、CRC-16/CCITT-FALSE 或自定义。校验放在报文最后一/两字节。默认无校验。

选择自定义时, 可设置字节序、多项式、取反、多项式、初始值和结果异或, 需有相关了解。

无

无

XOR(1Byte)

XOR,FF(1Byte)

SUM(1Byte)

CRC-16/MODBUS

CRC-16/CCITT(Kermit)

CRC-16/CCITT-FALSE

CRC-16/自定义

帧校验(CRC)

字节序

检查所有字节?

多项式

输入和输出按位取反?

初始值

结果异或(XOR)

CRC-16/自定义

大端模式(Big-Endian)

小端模式(Little-Endian)

0x10

0x21

0x0000

0xFFFF

0x0000

0xFFFF

若启用了校验, 可选择校验的范围。若不勾选检查所有字节, 需选择开始的字节位置 (包括它) 和结束的字节位置 (包括它)。

检查所有字节?

从报文第几个字节(包括它)开始校验?

从报文倒数第几个字节(包括它)结束校验?

1

2

(8) 485 帧发送间隔:

可选 20ms~1000ms 或禁用。默认禁用。禁用时, 两帧间发送间隔和波特率有关, 时间最低不低于 35bit(1bit 时间=1/波特率)和 2ms。

4.2 节点功能配置

1.4.1 KNX-485 转换模块 > 通道1 > 节点1

通用设置

通道1

节点1

节点2

节点3

通道2

节点1

节点2

通道3

通道4

启用

备注

报文

方向

读485设备模式?

数值遗传方式

启用发送时延

启用定时发送

485

485报文内容

KNX

DPT

数值(0/1)

启用

备注

KNX -> 485

485 -> KNX

全匹配(特定值)

部分匹配(计算值)

启用发送时延

启用定时发送

485

485报文内容

请输入16进制485报文。在部分匹配的模式下, 用'XX'表示数据可变的字节

KNX

DPT

数值(0/1)

[1.xxx] 1bit

0

1

4 / 18

- (1) 启用: 单独设置此转化节点是否启用
- (2) 备注: 可用于填写此节点报文的含义/功能

4.2.1 报文方向、读模式、透传模式、延时定时

➤ 参数说明

(1) 方向: 用于设置此报文的透传/转化方向。

○KNX->485:

表示设备在 KNX 总线接收 KNX 报文后, 经过一定规则转化后, 输出 485 报文发送到 485 总线上。主要三种情况: KNX 控 485 设备的控制报文、KNX 读 485 状态 (寄存器) 的读报文、KNX 受控设备状态变化告知 485 的状态反馈报文 (这种情况使用场合较少)。

○485->KNX:

表示设备在 485 总线接收 485 报文后, 经过节点的匹配和数据转化后, 输出 KNX 报文发送到 KNX 总线上。主要三种情况: 485 设备寄存器状态上报 (被读取或定时上报) 报文、485 设备控制 KNX 设备报文、485 设备主动读 KNX 设备 (模拟读寄存器, 这种情况使用场合较少)。

(2) 读 485 设备模式:

当方向为 KNX->485 时可勾选。勾选则表示这个节点/组对象在收到任意数据的 KNX 报文后, 都会触发发送 485 报文 (一条读指令)。搭配勾选发送延时和定时发送使用。

(3) 数据透传方式:

可选全匹配 (特定值) 和部分匹配 (计算值)。

○全匹配 (特定值):

适用所有 485 报文情况, 方向为 KNX->485 时, 当 KNX 总线收到报文数值与预设值相同时, 触发发送整条预设的 485 报文; 方向为 485->KNX 时, 当 485 总线收到报文与预设的 485 报文内容完全相同时 (KNX 的 DPT 为浮点型时, 精度误差内允许), 发送预设值的 KNX 报文。

○部分匹配 (计算值):

适配按一定规则的 485 报文, 若 485 报文数据段可变, 控制段固定, 可按规则透传。485 可变的字节用 'XX' 表示。数据段必须连续, 长度 1~4Bytes, 可在中间或末尾, 匹配时数据段可为任意值, 控制段必须完全相等。把 KNX 或 485 数据的值直接、或经计算后传到另一端。

(4) 启用发送时延: 勾选后, 报文满足条件触发发送时, 延时 0-255 秒后发送。

启用发送时延

☒

发送时延

3

秒(s)

(5) 启用定时发送: 勾选后, 报文满足条件触发发送时, 根据发送次数 (1~255 次, 0 表示始终发送) 和发送周期 (1~65535s) 发送。第一次立即发送, 后续发送的间隔即为发送周期时间。

若同时启用了发送时延和定时发送, 第一条发送的等待时间为设置的发送时延, 后续发送的间隔为设置的发送周期。

启用定时发送

☒

发送周期

3

秒

发送次数(0=始终发送)

3

4.2.2 报文内容、匹配透传方式、数据类型 DPT

➤ 参数说明

(1) 485 报文内容: 最长支持 30Bytes。数据透传方式选择 ☒ 全匹配(特定值) 时, 填入确定十六进制字节; 选择 ☐ 部分匹配(计算值) 时, 填入十六进制字节, 可变字节用 'XX' 代替。

忽略空格、大小写混用, 0x00~0x0F 不能缩写为 0x0~0xF。

全匹配示例如下:

485

485报文内容

55123 4 03 01 aD 8A

数据：

 55 12 34 03 01 AD 8A,Length=7

部分匹配示例如下：

485报文内容

32 1A 00 XX XX 33 06

数据：

 32 1A 00 XX XX 33 06,Length=7

● 合法性检查

数据库会检查输入的 485 文本是否合法，以下是常见的提示。

○当内容为空时，节点无效，被跳过，不进行任何匹配。提示如下

485报文内容

 请输入16进制485报文。在部分匹配的模式下，用'XX'表示数据可变的字节

○当报文长度超出 30 字节

485报文内容

\0B0C0D0E0F101112131415161718191A1B1C1D1E1F

 报文长度溢出

○当内容含有非 16 进制字符时，提示如下

485报文内容

32 1A 00 01 22 3u 06

 报文数据包含非16进制字符

○当数据字节不完整时，提示如下

485报文内容

32 1A 3 01 22 33

 数据无效,检查0x0~0xF是否已改为0x00~0x0F

○当全匹配模式下填入了可变字符'XX'，提示如下

485报文内容

32 1A xx 01 22 33

 数据无效,字符X只能在"部分匹配"模式

○当部分匹配模式下缺少可变字符'XX'，提示如下

485报文内容

32 1A 32 01 22 33

✖ 部分匹配模式下，485报文中可变字段为空，请用XX填充每个可变的字节

○当错误地使用了'X'时

485报文内容

32 12 01 Fx 33

✖ 数据无效，请检查'X'

○当部分匹配模式下可变字节长度超过 4 字节时

485报文内容

32 12 01 xx xx xx xx xx 33

✖ 485报文中连续可变字段的长度必须在1~4字节

(2) DPT: 选择 KNX 的报文数据类型。

● 常见数据点

类型	常用适用范围
1bit[1.xxx]开关类型	开关、布尔逻辑、上下、电平高低等
4bit[3.007]相对调光	相对调光
1Byte[5.001]百分比类型	0-100%范围的调光、窗帘位置百分比、风速百分比等
1Byte[5.010]挡位/计数	0-255 范围的调光、风速挡位、模式码
1Byte[18.001]场景控制	支持场景号控制
2Byte[7.xxx]计数	0-65535 范围的计数、时间、长度、电流、lux 亮度、色温等
2Byte[9.xxx]浮点型	温度、湿度、气压、CO2 浓度、速度、功率、电流电压等传感器参数
4Byte[12.xxx]计数	大范围整数的计数、时间、体积等
4Byte[13.xxx]浮点型	高精度的加速度、面积、能量、电容、电阻、温度等物理参数

注: 485 报文的数据类型和 KNX 可能不一致, 如某些 485 设备用两字节的 FFFF 表示开 0000 表示关, 对 KNX 设备来说开关通常为 1bit 的 0 关 1 开。

[1.xxx] 1bit

[1.xxx] 1bit

[3.007] 4bit:相对调光

[5.001] 1Byte:百分比 (0..100%)

[5.010] 1Byte:计数 (0..255)

[18.001] 1Byte:场景控制(1...64)

[7.xxx] 2-Byte 无符号(0...65535)

[9.xxx] 2-Byte 浮点数

[12.xxx] 4-Byte 无符号

[13.xxx] 4-Byte 浮点数

- 全匹配, 双向, KNX<->485

(3)数值

- [1.xxx] 1bit: 可选 0/1

DPT [1.xxx] 1bit

数值(0/1) ☒ 0 ☐ 1

- [3.007] 4bit 相对调光: 可选停止调光, 增加/减少 100%/50%/25%/12.5%/6.25%

DPT [3.007] 4bit:相对调光

数值(4bit)

停止减少

停止减少

-100%

-50%

-25%

-12.50%

-6.25%

-3.13%

-1.56%

停止增加

+100%

+50%

+25%

+12.5%

+6.25%

+3.13%

+1.56%

- [5.001] 1Byte 百分比: 可设置 0-100%的值

DPT [5.001] 1Byte:百分比 (0..100%)

数值(0...100%) 0 %

- [5.010] 1Byte 计数: 可设置 0-255 的值

DPT [5.010] 1Byte:计数 (0..255)

数值(0...255) 0

- [18.001] 1Byte 场景控制: 可设置激活/学习场景, 场景号 1~64

DPT [18.001] 1Byte:场景控制(1...64)

场景号 激活/场景42

- [7.xxx] 2Byte 无符号: 可设置 0-65535 的值

DPT [7.xxx] 2-Byte 无符号(0...65535)

数值(0...65535) 3000

- [9.xxx] 2Byte 浮点数: 可设置-6700760~670760 的值
方向为 KNX->485 时, 需选择取整方式: 四舍五入、向上/向下取整。

DPT	[9.xxx] 2-Byte 浮点数
取整方式	四舍五入
数值(2Byte float)	-670760 ... 670760

○ [12.xxx] 4Byte 无符号: 可设置 $0 \sim 2^{32}-1$ 的值

DPT	[12.xxx] 4-Byte 无符号
数值(4Byte unsigned)	0

○ [13.xxx] 4Byte 浮点数: 可设置 $-3.4E+38 \sim 3.4E+38$ 的值

DPT	[13.xxx] 4-Byte 浮点数
取整方式	四舍五入
数值(4Byte float)	20.5

-3.402823E+38 ... 3.402823E+38

4.2.3 部分匹配的逻辑计算功能

4.2.3.1 KNX->485 方向

➤ 参数说明

(1) 启用计算 1/启用计算 2: 表示是否对 KNX 数据进行逻辑运算。

都不启用时, 数值直接透传。同时启用计算 1 和 2 时, 先对原始数据与 value1 进行运算得到 out1, 再把 out1 和 value2 进行运算得到最终数据结果 out2。

(2) 计算 1/2 和 value1/2

不同 DPT 运算方式不同

○ [1.xxx] [5.001] [5.010] [18.001] [7.xxx] [12.xxx]

启用计算1	☒
计算1	OUT1 = knxdata + value1
value1	OUT1 = knxdata
启用计算2	OUT1 = knxdata + value1
	OUT1 = knxdata - value1
	OUT1 = value1 - knxdata
	OUT1 = knxdata * value1
	OUT1 = knxdata / value1
	OUT1 = 1 << (knxdata + value1)
	OUT1 = 1 << (knxdata - value1)
	OUT1 = 1 << (value1 - knxdata)
计算2	OUT2 = OUT1 + value2
value2	OUT2 = OUT1
	OUT2 = OUT1 + value2
	OUT2 = OUT1 - value2
	OUT2 = value2 - OUT1
	OUT2 = OUT1 * value2
	OUT2 = OUT1 / value2
	OUT2 = 1 << (OUT1 + value2)
	OUT2 = 1 << (OUT1 - value2)
	OUT2 = 1 << (value2 - OUT1)

整型的值与 value1/2 (4 字节无符号) 进行直接赋值、加减乘除、左移操作

○ [9.xxx] [13.xxx]

浮点型的值与 value1/2 进行直接赋值、加减乘除, 然后取整, value1/2 精度为 0.01

计算1	OUT1 = knxdata * value1
value1	OUT1 = knxdata
启用计算2	OUT1 = knxdata + value1
计算2	OUT1 = knxdata - value1
value2	OUT1 = value1 - knxdata
	OUT1 = knxdata * value1
	OUT1 = knxdata / value1

value1	5.5
--------	-----

注: knxdata 即为总线上收到报文根据协议对应的 DPT 的解析值

4.2.3.2 485->KNX 方向

➤ 参数说明

(1) 启用计算 1/启用计算 2: 表示是否对 485 数据进行逻辑运算。

485 数据在 1~4 字节。都不启用时, 数值直接透传。先对原始数据与 value1 进行运算得到 out1, 再把 out1 和 value2 进行运算得到最终数据结果 out2。部分类型只有计算 1。

(2) 计算 1/2 和 value1/2

VALUE1/2 长度根据 485 字节长度变化

○ [1.xxx]

DPT	[1.xxx] 1bit
启用计算1	☒
计算1	OUT1 = 1(XX>=value1), 0(XX<value1)
VALUE1	0x00
	OUT1 = 1(XX>0), 0(XX=0)
	OUT1 = 1(XX=0), 0(XX>0)
	OUT1 = 1(XX>=value1), 0(XX<value1)
	OUT1 = 1(XX<value1), 0(XX>=value1)

VALUE1	0x00	0x00	0x00
--------	-----------------------------------	-----------------------------------	-----------------------------------

485 数据≥Value1 (1~4 字节) 或>0 时, KNX 数据取 0/1。

○ [3.007]

DPT	[3.007] 4bit:相对调光
启用计算1	☒
计算1	OUT1 = (XX[LSB] + value1) & 0xF
VALUE1	
	OUT1 = XX[LSB] & 0xF
	OUT1 = (XX[LSB] + value1) & 0xF
	OUT1 = (XX[LSB] - value1) & 0xF
	OUT1 = (value1 - XX[LSB]) & 0xF

485 数据最低字节与 value1 (1 字节) 进行加减偏移, 取最后 4bit。

○ [5.001] [5.010] [7.xxx] [12.xxx]

启用计算1 ☒

计算1

VALUE1

启用计算2

计算2

VALUE2

OUT1 = XX * value1

OUT1 = XX

OUT1 = XX + value1

OUT1 = XX - value1

OUT1 = value1 - XX

OUT1 = XX * value1 ☒

OUT1 = XX / value1

OUT1 = log2(XX) + value1

OUT1 = log2(XX) - value1

OUT1 = value1 - log2(XX)

485 数据与 value1/2 (1/2/4 字节) 进行直接赋值、加减乘除、取对数, 取相应字节。

○ [18.001]

DPT

启用计算1 ☒

计算1(场景)

VALUE1

Active, Scene NO = XX + value1

Active, Scene NO = XX

Active, Scene NO = XX + value1 ☒

Active, Scene NO = XX - value1

Active, Scene NO = value1 - XX

Learn, Scene NO = XX

Learn, Scene NO = XX + value1

Learn, Scene NO = XX - value1

Learn, Scene NO = value1 - XX

485 数据最低字节与 value1 进行加减偏移, 作为 KNX 激活/学习的场景号

○ [9.xxx] [13.xxx]

启用计算1 ☒

计算1

value1

启用计算2

计算2

value1

计算2

OUT1 = int(485data) * value1

OUT1 = int(485data)

OUT1 = int(485data) + value1

OUT1 = int(485data) - value1

OUT1 = value1 - int(485data)

OUT1 = int(485data) * value1 ☒

OUT1 = int(485data) / value1

OUT2 = OUT1

OUT2 = OUT1 + value2

OUT2 = OUT1 - value2

OUT2 = value2 - OUT1

OUT2 = OUT1 * value2

OUT2 = OUT1 / value2

485 数据转化为有符号的数后, 与 value1/2 (浮点型) 进行直接赋值、加减乘除, 转化为浮点型的 KNX 数据。

5 配置示例

例 1: 某 485 设备开关机 485

无校验, 用 KNX 设备控制它, 采用部分匹配, 一个节点(id:1)连接群组地址 0/3/1

1

CH1_Object1

KNX->485:1bit

KNX->485:1bit

0/3/1

1 bit

C - W - - switch

备注

示例1：控制设备开关机

报文

方向

☒ KNX -> 485

☐ 485 -> KNX

读485设备模式?

☐

数值透传方式

☐ 全匹配(特定值)

☒ 部分匹配(计算值)

(id:1)

1) 开机: 0A 33 12 **00 00**,关机: 0A 33 12 **01 01**
关(knx:0 对应 485: 0x0101),开 (knx:1 对应 485: 0x0000)

485报文内容

0A 33 12 XX XX

DPT

[1.xxx] 1bit

启用计算1

☒

计算1

OUT1 = knxdata * value1

value1

257

2) 开机: 0A 33 12 **40**,关机: 0A 33 12 **80**
关 0x80(knx:0 对应 485:1<<7), 开 0x40(knx:1 对应 485:1<<6)

485报文内容

0A 33 12 XX

DPT

[1.xxx] 1bit

启用计算1

☒

计算1

OUT1 = 1 << (value1 - knxdata)

value1

7

3) 开机: 0A 33 12 **41**,关机: 0A 33 12 **40**
关 (knx:0 对应 485: 0x40), 开(knx:1 对应 485: 0x41)

DPT

[1.xxx] 1bit

启用计算1

☒

计算1

OUT1 = knxdata + value1

value1

64

KNX 总线收到其他 KNX 设备的报文

Time	Source Addr	Destination A	Hop	Type	DPT	Info
2026-01-08 14:55:14.646	1.0.65	0/3/1	6	GroupValue_Write	1.001 switch	\$00 Off
2026-01-08 14:55:20.462	1.0.65	0/3/1	6	GroupValue_Write	1.001 switch	\$01 On

转化模块往 485 总线发报文 (RECV 表示串口调试工具收到)

```
[2026-01-08 14:55:14.715]# RECV HEX/4 <<<
0A 33 12 40
[2026-01-08 14:55:20.535]# RECV HEX/4 <<<
0A 33 12 41
```

例 2: 某 485 设备控制新风挡位
(SUM 校验), 采用部分匹配, 一个节点(id:2)连接群组地址 0/3/2

2	CH1_Object2	KNX->485:1Byte(0...255)	KNX->485:1Byte(0...255)	0/3/2	1 byte	C - W - -	counter pulses (0..255)
---	-------------	-------------------------	-------------------------	-------	--------	-----------	-------------------------

备注

示例2：控制设备新风挡位

报文

方向

☒ KNX -> 485 ☐ 485 -> KNX

读485设备模式？

☐

数值透传方式

☐ 全匹配(特定值) ☒ 部分匹配(计算值)

(id:2)

关闭: 01 34 08 01 01 00 3F;
低速: 01 34 04 01 01 00 3b;
中速: 01 34 02 01 01 00 39
高速: 01 34 01 01 01 00 38
KNX 的低中高三档挡位值通常为 01/02/03 (一字节)

485报文内容

01 34 XX 01 01

低速 04(knx:01 对应 485: 1<<2), 中速 02(knx:02 对应 485: 1<<1), 高速 01(knx:02 对应 485: 1<<0)
先设置 CRC 校验, 校验字节不放入 485 报文内容中

帧校验(CRC)

SUM(1Byte)

检查所有字节？

☒

DPT

[5.010] 1Byte:计数 (0..255)

启用计算1

☒

计算1

OUT1 = 1 << (value1 - knxdata)

value1

3

KNX 总线收到其他 KNX 设备的报文

2026-01-08 16:13:10.556	1.0.65	0/3/2	6	GroupValue_Write	5.010 counter pulses (0.....	\$01 1 counter pulses
2026-01-08 16:13:12.264	1.0.65	0/3/2	6	GroupValue_Write	5.010 counter pulses (0.....	\$02 2 counter pulses
2026-01-08 16:13:14.572	1.0.65	0/3/2	6	GroupValue_Write	5.010 counter pulses (0.....	\$03 3 counter pulses
2026-01-08 16:13:18.344	1.0.65	0/3/2	6	GroupValue_Write	5.010 counter pulses (0.....	\$00 0 counter pulses

转化模块往 485 总线发报文 (RECV 表示串口调试工具收到)

```
[2026-01-08 16:13:10.623]# RECV HEX/6 <<<
01 34 04 01 01 3B
[2026-01-08 16:13:12.337]# RECV HEX/6 <<<
01 34 02 01 01 39
[2026-01-08 16:13:14.646]# RECV HEX/6 <<<
01 34 01 01 01 38
[2026-01-08 16:13:18.409]# RECV HEX/6 <<<
01 34 08 01 01 3F
```

例 3: 某 485 调光设备设置亮度和读状态
(SUM 校验), 用 KNX 设备控制它, 转化模块需要在控制后的一段时间后往 485 总线读取设备亮度寄存器的状态, 设备读响应后把亮度状态值上报到 KNX 总线, 需要配置 3 个节点(id:3、4、5)。
设置亮度: 01 30 XX 01 01 00 CRC (XX 为亮度百分比, 如 35%亮度为 01 30 23 01 01 00 56)
读亮度寄存器指令: 03 30 A0 23 00 01 01 00 CRC
亮度寄存器读响应: 07 30 00 01 XX 00 CRC
控制和读响应报文部分匹配, 读报文全匹配。三个节点配置如下, 注意读模式、报文延时和方向。

备注

示例3：调光设备亮度设置

报文

方向

☒ KNX -> 485 ☐ 485 -> KNX

读485设备模式？

☐

数值透传方式

☐ 全匹配(特定值) ☒ 部分匹配(计算值)

启用发送时延

☐

启用定时发送

☐

485

485报文内容

01 30 XX 01 01 00

数据：

01 30 XX 01 01 00,Length=6

KNX

DPT

[5.001] 1Byte:百分比 (0..100%)

启用计算1

☒

计算1

OUT1 = knxdata (id:3)

备注 示例3：读调光设备亮度

报文

方向 ☒ KNX -> 485 ☐ 485 -> KNX

读485设备模式？ ☒

数值透传方式 全匹配(特定值)

启用发送时延 ☒

发送时延 5 秒(s)

启用定时发送 ☐

485

485报文内容 03 30 A0 23 00 01 01 00

数据：

 03 30 A0 23 00 01 01 00,Length=8

KNX

DPT [5.001] 1Byte:百分比 (0..100%) (id:4)

方向 ☐ KNX -> 485 ☒ 485 -> KNX

数值透传方式 ☐ 全匹配(特定值) ☒ 部分匹配(计算值)

启用发送时延 ☐

启用定时发送 ☐

485

485报文内容 07 30 00 01 XX 00

数据：

 07 30 00 01 XX 00,Length=6

KNX

DPT [5.001] 1Byte:百分比 (0..100%)

启用计算1 ☒

计算1 OUT1 = XX (id:5)

	3	CH1_Object3	KNX->485:百分比(0-100%)	亮度控制	KNX->485:百分比(0-100%)	0/3/3	1 byte	C	-	W	-	-	percentage (0..100%)
	4	CH1_Object4	KNX->485:百分比(0-100%)	读	KNX->485:百分比(0-100%)	0/3/3	1 byte	C	-	W	-	-	percentage (0..100%)
	5	CH1_Object5	485->KNX:百分比(0-100%)	亮度状态反馈	485->KNX:百分比(0-100%)	0/3/4	1 byte	C	R	-	T	-	percentage (0..100%)

设置亮度和读亮度两个节点均绑定 0/3/3, 读响应节点绑定 0/3/4。转化模块收到 KNX 亮度调节报文后立即往 485 发送设置亮度、5 秒后读取亮度寄存器, 随后 485 设备上报亮度状态。

KNX 总线和 485 捕获报文如下

Time	Source Addr	Destination Addr	Hop	Type	DPT	Info
2026-01-08 17:33:10.611	1.0.65	0/3/3	6	GroupValue_Write	5.001 percentage (0..100%)	\$66 40.0%
2026-01-08 17:33:15.691	1.4.1	0/3/4	6	GroupValue_Write	5.001 percentage (0..100%)	\$66 40.0%

[2026-01-08 17:33:10.678]# RECV HEX/7 <<<
01 30 28 01 01 00 58
[2026-01-08 17:33:15.623]# RECV HEX/9 <<<
03 30 A0 23 00 01 01 00 F8
[2026-01-08 17:33:15.654]# AUTO REPLY HEX/7 to COM24 >>>
07 30 00 01 28 00 60

控制
读
响应

第一条 KNX 报文表示其他 KNX 设备发出一条设置亮度为 40%的报文(KNX#1), 经过转化模块后 (同时触发群组 id3 的控制和 id4 的读), 立即往 485 总线发出控制(485#1), 5 秒后再往 485 总线发出读(485#2), 485 设备再响应(485#3)当前亮度 0x28=40%, 经过转化模块后往 KNX 总线上报状态 (KNX#2)。

例 4: 某 485 温控设备设置温度和读状态

CCITT16-MODBUS 校验, 大端模式, 用 KNX 设备控制它, 转化模块需要在控制后的一段时间后往 485 总线读取设备温度寄存器的状态, 设备读响应后, 转化模块把温度状态值上报到 KNX 总线, 需要配置 3 个节点(id:6、7、8)。

- 设置温度: 01 42 03 XX XX CRC CRC (XXXX 为带符号的两字节,uint16)
- 读温度寄存器: 03 0F 1A 42 03 00 02 84 14
- 状态上报: 06 DC 42 03 XX XX CRC CRC

备注

示例4：设置温度

报文

方向

☒ KNX -> 485

☐ 485 -> KNX

读485设备模式？

☐

数值透传方式

☐ 全匹配(特定值)

☒ 部分匹配(计算值)

启用发送时延

☐

启用定时发送

☐

485

485报文内容

01 42 03 XX XX

数据：

01 42 03 XX XX,Length=5

KNX

DPT

[9.xxx] 2-Byte 浮点数

取整方式

四舍五入

启用计算1

☒

计算1

OUT1 = knxdata * value1

value1

10

(id:6)

16 / 18

备注 示例4：读温度寄存器

报文

方向 ☒ KNX -> 485 ☐ 485 -> KNX

读485设备模式? ☒

数值透传方式 全匹配(特定值)

启用发送时延 ☒

发送时延 3 秒(s)

启用定时发送 ☒

发送周期 5 秒

发送次数(0=始终发送) 2

485

485报文内容 03 0F 1A 42 03 00 02

数据：

 03 0F 1A 42 03 00 02,Length=7

KNX

DPT [9.xxx] 2-Byte 浮点数 (id:7)

报文

方向 ☐ KNX -> 485 ☒ 485 -> KNX

数值透传方式 ☐ 全匹配(特定值) ☒ 部分匹配(计算值)

启用发送时延 ☐

启用定时发送 ☐

485

485报文内容 06 DC 42 03 XX XX

数据：

 06 DC 42 03 XX XX,Length=6

KNX

DPT [9.xxx] 2-Byte 浮点数

启用计算1 ☒

计算1 OUT1 = int(485data) / value1

value1 10 (id:8)

帧校验(CRC)

CRC-16/MODBUS

字节序

☒ 大端模式(Big-Endian)
 ☐ 小端模式(Little-Endian)

检查所有字节?
 ☒

42 03	xx xx	Room temperature (real*10) (-41°C ~ 100°C) Ex) 0x0000→0°C, 0xFFFF→-1→-0.1°C 0xFFF6→-10→-1°C, 0xFE66→-410→-41.0°C 0x03E8→1000→100°C
-------	-------	--

解析:

设置温度时, 485 数据段的温度值为实际值*10, 精确到 0.1°C。485 模块收到 KNX 总线的设置温度报文 26.52°C->乘 10 取整为+265->摄氏度设置温度 26.5°C(265 = 0x0109)->转化为 485 数据: 01 42 03 01 09 2E 3C, 设置温度-5.3°C->-53(0xFFCB)为 01 42 03 FF CB DF FD。

读取温度在触发设置温度后的 3s 后发送, 定时读取两次, 定时周期为 5s。

温度状态上报如 485 上报 06 DC 42 03 01 09->数据段 0x0109 值为 265->除以 10 表示温度为 26.5°C, 转发到 KNX 总线。

6	CH1_Object6	KNX->485:2Byte(-32768...32767)	KNX->485:2Byte(-32768...3... 0/3/5	2 bytes	C	-	W	-	-	2-byte float value
7	CH1_Object7	KNX->485:2Byte(-32768...32767)	KNX->485:2Byte(-32768...3... 0/3/5	2 bytes	C	-	W	-	-	2-byte float value
8	CH1_Object8	485->KNX:2Byte(-32768...32767)	485->KNX:2Byte(-32768...3... 0/3/6	2 bytes	C	R	-	T	-	2-byte float value

KNX 总线和 485 捕获报文如下

[2026-01-09 15:05:00.512]# RECV HEX/7 <<<
 01 42 03 01 09 2E 3C
 [2026-01-09 15:05:03.467]# RECV HEX/9 <<<
 03 0F 1A 42 03 00 02 84 14
 [2026-01-09 15:05:03.512]# AUTO REPLY HEX/8 to COM24 >>>
 06 DC 42 03 01 09 40 F4
 [2026-01-09 15:05:08.461]# RECV HEX/9 <<<
 03 0F 1A 42 03 00 02 84 14
 [2026-01-09 15:05:08.493]# AUTO REPLY HEX/8 to COM24 >>>
 06 DC 42 03 01 09 40 F4

#	Time *	Source Add	Destination A	Hop	Type	DPT	Info
7	2026-01-09 15:05:00.444	1.0.65	0/3/5	6	GroupValue_Write	9.* 2-byte float value	0D 2E 26.52
8	2026-01-09 15:05:03.548	1.4.1	0/3/6	6	GroupValue_Write	9.* 2-byte float value	0D 2D 26.5
9	2026-01-09 15:05:08.532	1.4.1	0/3/6	6	GroupValue_Write	9.* 2-byte float value	0D 2D 26.5

转化模块在 KNX 总线收到设置温度 26.52°C指令 (KNX#1)->转化后往 485 总线发送设置温度为 26.5°C指令 (485#1)->3 秒后往 485 总线读温度寄存器指令 (485#2)->485 温控设备在 485 总线回复当前温度值 26.5 摄氏度 (485#3)->转化模块把 485 总线的温度的读回复数据转化后抛到 KNX 总线 (KNX#2)->转化模块定时 5 秒后再次往 485 总线读温度寄存器指令 (485#4)->485 温控设备在 485 总线回复当前温度值 26.5 摄氏度 (485#5)->转化模块再次把 485 总线的温度的读回复数据转化后抛到 KNX 总线 (KNX#3)