

# Pós-Graduação em Infra-Estruturas de Telecomunicações Segurança e Domótica



**TELEGRAMA**

Eng.º Domingos Salvador dos Santos

email:dss@isep.ipp.pt

Setembro de 2007



## Estrutura da Apresentação

- **Formato de Dados KNX**
- **Sinal no Barramento**
- **Acesso ao Meio**
- **Características do Telegrama**
- **Campos do Telegrama**
- **Reconhecimento do Telegrama**
- **Modelo OSI e Modelos EIS**

## Formato de Dados KNX

### ➤ Sistemas Numéricos

Sistema Numérico	DECIMAL	BINÁRIO	HEXADECIMAL
Base	10	2	16
Digitos	0,1,2,...,9	0,1	0,1,...,9,A,B,...F

### ➤ Sistema Decimal:

- Este sistema é o mais comum dos sistemas numéricos. As pessoas pensam em termos de números decimais.

## Formato de Dados KNX

### ➤ Sistema Binário:

- Esta representação numérica é muito importante em computação, uma vez que o hardware de um computador apenas interpreta dois estados (0,1). A cada espaço de memória é chamado um bit.

### ➤ Sistema Hexadecimal:

- Uma combinação de quatro bits de um sistema binário resulta num número hexadecimal. Este sistema oferece uma clara representação da informação e dos dados.

# Formato de Dados KNX

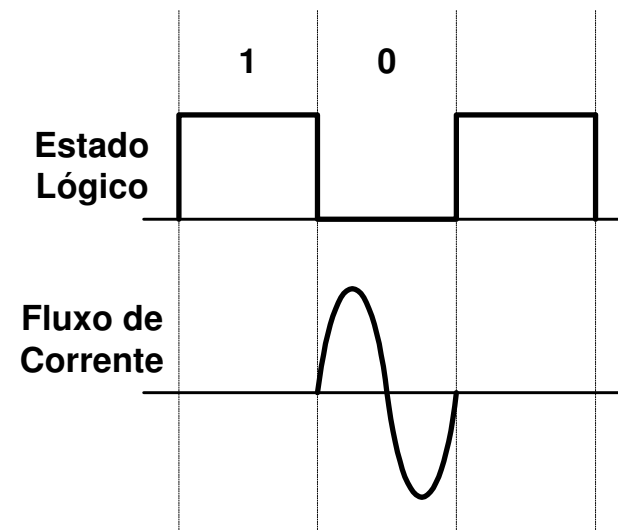
Tamanho da Informação	Valores Representáveis	Termo em Tecnologia Digital	Aplicação KNX (Seleção)
1 Bit	2	Bit	Ligar/Desligar
2 Bits	4		Prioridade
4 Bits	16	Grupo de quatro	Regulação de Fluxo
8 Bits	256	Byte	Analizador
16 Bits	65536	Palavra	Pontos Flutuantes
32 Bits	4294967296	Palavra Dupla	Contador

## Sinal no Barramento

### ➤ No KNX:

- Estado Lógico “1” → Não flui corrente
- Estado Lógico “0” → Flui corrente

➤ Se vários participantes transmitem simultaneamente, o estado lógico “0” prevalece.



## Acesso ao Meio

- O acesso ao meio é baseado no protocolo CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*), que funciona da seguinte forma:
  - Um dispositivo que pretende transmitir uma mensagem pode começar a fazê-lo imediatamente, se encontrar o barramento desocupado (inactivo). Caso contrário, terá de esperar até este ficar livre;
  - Os dispositivos escutam o barramento enquanto transmitem, com o objectivo de detectarem a colisão, que ocorrerá se dois dispositivos transmitirem em simultâneo;
  - Quando o dispositivo que tenta impor o estado lógico "1" detecta o estado lógico "0" (sinal de corrente na linha), pára de transmitir para permitir que o dispositivo com a mensagem de mais alta prioridade o continue a fazer;

## Acesso ao Meio

### ➤ Protocolo CSMA/CA (continuação):

- O dispositivo com a mensagem de baixa prioridade continua no entanto a escutar o barramento, de modo a que quando terminar a transmissão, possa então transmitir os seus dados.

Mensagem 1

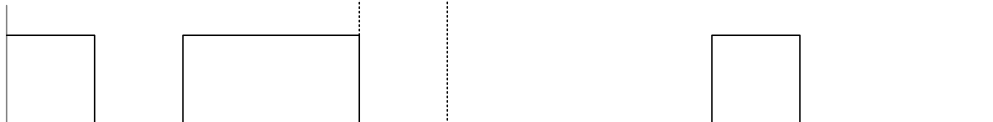


Mensagem 2



A transmissão da mensagem 2 é abortada  
(repete ao fim de um determinado tempo)

Barramento

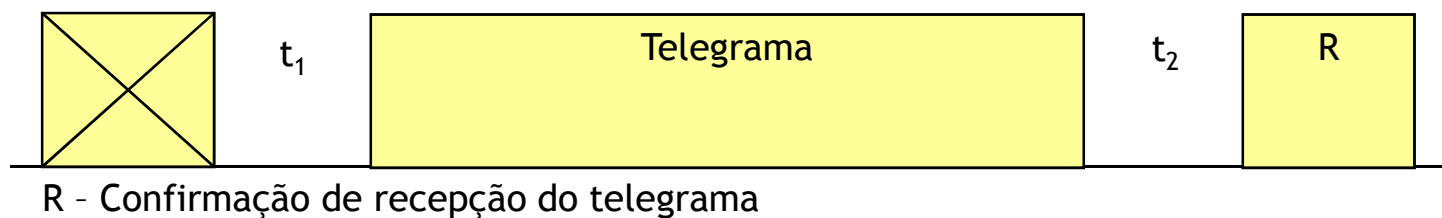




## Características do Telegrama

### ➤ Generalidades

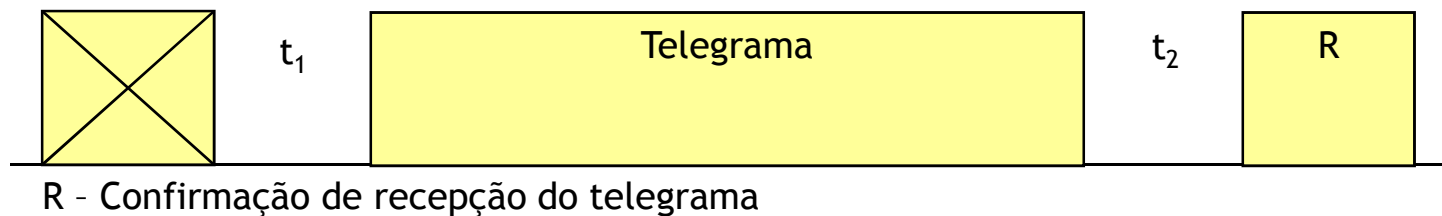
- Quando ocorre uma ordem (ex. quando um botão de pressão é carregado), o participante envia um telegrama para a linha de bus.
- A transmissão começa depois da linha permanecer desocupada por um período não inferior a  $t_1$ .



## Características do Telegrama

### ➤ Generalidades

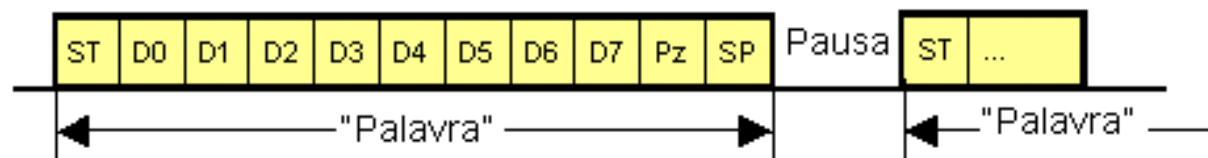
- Uma vez completa a transmissão do telegrama, o participante usa o período  $t_2$  para verificar se o telegrama chegou nas melhores condições.
- Todos os participantes endereçados tomam conhecimento da recepção do telegrama simultaneamente.



## Características do Telegrama

### ➤ Caracteres do Telegrama

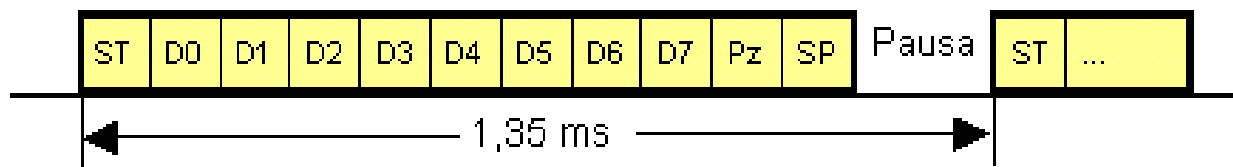
- Os telegramas são transmitidos de uma forma assíncrona. Bits de arranque e de paragem são necessários para a sincronização e recepção dos participantes.
- Cada caracter que é transmitido representa um grupo de 8 bits de informação (= 1 byte).
- O Bit de arranque (ST-Start Bit) precede o 1º bit do caracter.



## Características do Telegrama

### ➤ Tempo Requerido por um Telegrama

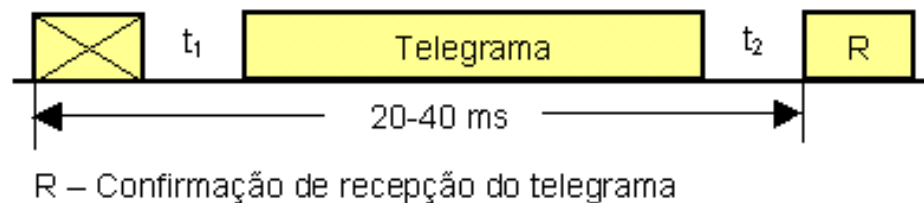
- O telegrama é transmitido a uma taxa de 9600 bits/s, o que significa que um bit ocupa a linha de comunicação por um período de  $1/9600$ s ou  $104\mu\text{s}$ .
- O caracter consiste em 11 bits. Somando o intervalo que decorre antes do próximo caracter, decorre um total de um período de 13 bits ou  $1,35\text{ms}$ .



## Características do Telegrama

### ➤ Tempo Requerido por um Telegrama

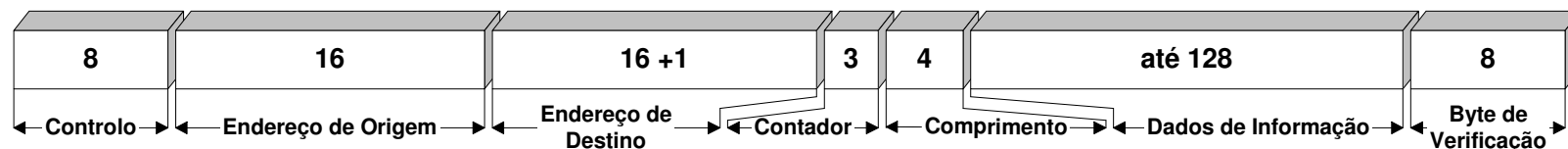
- O telegrama pode ter de 8 a 23 caracteres de tamanho, dependendo da quantidade da informação. O reconhecimento apenas contém um caracter.
- Tendo em consideração o período livre da linha ( $t_1=50\text{bits}$ ) e o tempo que decorre antes da transmissão do reconhecimento ( $t_2=13\text{ bits}$ ), um telegrama ocupa a linha de 20 a 40ms.
- Os telegramas mais comuns ocupam a linha de bus durante 20ms.



## Campos do Telegrama

### ➤ Estrutura do Telegrama

- O telegrama consiste em informação específica da linha, fornecendo os dados e as características da ordem.
- A informação é transmitida na sua globalidade sob a forma de caracteres de 8 bits.



## Campos do Telegrama

### ➤ Campo de Controlo

Campo de Controlo								
D7	D6	D5	D4	D3	D2	D1	D0	
1	0	R	1	P1	P2	0	0	Prioridade de Transmissão
				0	0			Funções de Sistema (Prioridade Máxima)
				1	0			Funções de Alarme
				0	1			Prioridade de Serviço Elevada (Manual)
				1	1			Prioridade de Serviço Baixa (Automático)
		0						Repetição de Telegrama
		1						Telegrama Normal

- Se um dos participantes endereçados responde negativamente ao reconhecimento, a transmissão do telegrama é repetida com o bit D5 do campo de controlo a “0”.

## Campos do Telegrama

### ➤ Campo de Controlo

- Neste caso é assegurado que o participante que já tenha executado o comando respectivo não o repita novamente.
- A prioridade da transmissão é apenas considerada se vários participantes tentarem transmitir simultaneamente.
- A prioridade requerida (à parte das funções do sistema) pode ser definida para todos os objectos de comunicação através do EIB TOOL SOFTWARE (ETS). As definições standard apresentam baixa prioridade operacional.

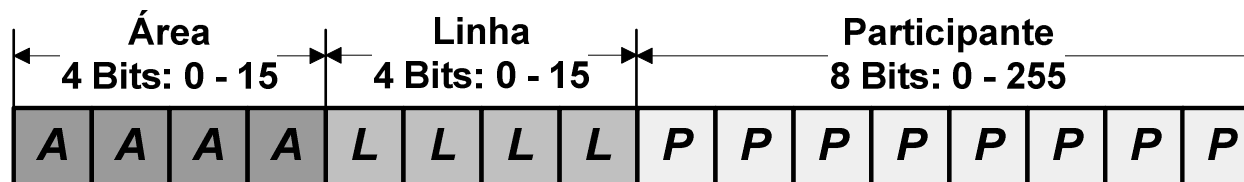


## Campos do Telegrama

### ➤ Endereço de Origem

- O endereço de origem é o endereço físico do participante que transmite.
- Quando enviado, o participante fornece o endereço físico juntamente com o telegrama, de tal modo que este possa ser facilmente reconduzido à sua origem.

#### Endereço Físico

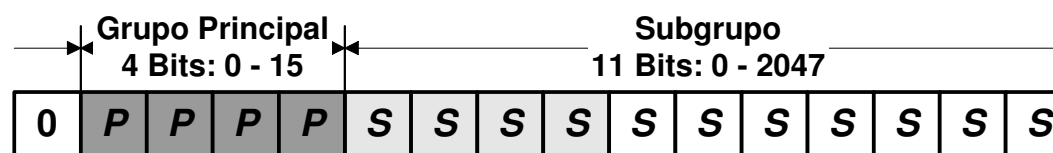


## Campos do Telegrama

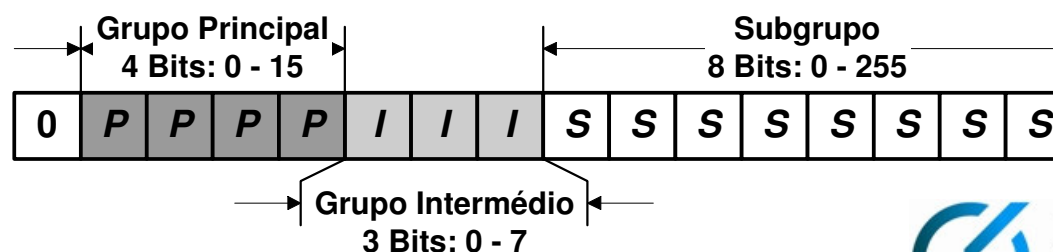
### ➤ Endereço de Destino

- O endereço de destino especifica que participante deve reagir ao telegrama. O endereço de destino é normalmente um endereço de grupo, com o qual um número ilimitado de participantes podem ser endereçados simultaneamente.

#### Endereços de grupo com 2 níveis



#### Endereços de grupo com 3 níveis



## Campos do Telegrama

### ➤ Endereço de Destino

- O endereço de destino pode ser também um endereço físico. Isto acontece sempre que estamos no modo de inicialização do sistema, programação e diagnóstico.
- Assim, a informação é transmitida em 17 bits por forma a que o receptor saiba qual o tipo de endereço de destino.

Endereço de Destino								Endereço de Destino								Tipo de Endereço de Destino	
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0	D7	
A3	A2	A1	A0	L3	L2	L1	L0	P7	P6	P5	P4	P3	P2	P1	P0	0	Endereço Físico
R	P3	P2	P1	P0	I2	I1	I0	S7	S6	S5	S4	S3	S2	S1	S0	1	Endereço de Grupo

## Campos do Telegrama

### ➤ Campo Contador e Comprimento (D/R/L)

- O telegrama transmitido pelo participante contém um contador de voltas inicializado com o valor 6.
- Cada acoplador de linha decrementa o contador e transmite o telegrama enquanto o valor seja positivo. As tabelas de filtragem são tidas sempre em linha de conta.

Campo D/R/L							
D7	D6	D5	D4	D3	D2	D1	D0
D	R2	R1	R0	L3	L2	L1	L0
				X	X	X	X
0							
1							

Comprimento do Campo de Dados

Contador de Informação

Endereço Físico

Endereço de Grupo

## Campos do Telegrama

### ➤ Campo de Dados

- A maioria dos telegramas transmite apenas uma ordem de 1 bit.
- A ordem "escrever" corresponde ao último bit da direita. Coloca-se "1" ou "0", para activação ou desactivação, respectivamente. A informação útil tem 2 Bytes (Byte 0 e 1) de comprimento.

Comprimento				Ordem				Parâmetros																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																	
L	L	L	L	0	0	X	X	X	X	B	B	B	B																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																												





X = Não é utilizado

C = Confirmação

# Campos do Telegrama

## Campo de Dados

- Com a ordem "Ier" é solicitado ao dispositivo que recebe o telegrama um aviso do seu estado.
- A informação pode ter um comprimento de 1 bit ou pode utilizar até 13 Bytes (Byte 2 a 15).
- O tamanho da informação depende do tipo de EIS usado.

Comprimento				Ordem												Parâmetros																			
L	L	L	L	0	0	X	X	X	X	B	B	B	B													...									
0	0	0	1	Escrever				0	0	1	0	0	0	0	0	1																			
0	0	0	1	Ler				0	0	0	0	X	X	X	X	X	X																		
0	0	0	1	Confirmação curta				0	0	0	1	C	C	C	C	C	C																		
1	1	1	1	Confirmação longa				0	0	0	1	X	X	X	X	X	X	C	C	C	C	C	C	C	C	...	C	C	C	C	C	C	C		
																																			
				Byte 0								Byte 1								Byte 2								Byte 15							

X = Não é utilizado  
C = Confirmação

## Campos do Telegrama

### ➤ Campo de Verificação

- O telegrama também possui um campo de verificação, que confirma se o telegrama chega correctamente ao seu destino ou não.
- Esta parte do telegrama comprova se as restantes "palavras" que o constituem estão bem construídas ou não.
- Verificação de Caracter - Para a confirmação da "palavra", a soma dos bits de dados (D7 a D0) e de Pz deve ser de 0. Chama-se a isto confirmação por paridade par.

D7	D6	D5	D4	D3	D2	D1	D0	+	Pz	=	0
----	----	----	----	----	----	----	----	---	----	---	---

## Campos do Telegrama

### ➤ Campo de Verificação

- Verificação de Telegrama - A posição dos bits de todos os caracteres do telegrama são verificadas por prioridade, isto é, o bit de verificação S7 recebe o valor “0” ou “1” para fazer a soma de todos os bits de dados D7 mais os bits de verificação S7 iguais a 1.
- A combinação da verificação dos caracteres e do telegrama é chamada Verificação Cruzada.

D7	D6	D5	D4	D3	D2	D1	D0	+	Pz	=	0
D7	D6	D5	D4	D3	D2	D1	D0	+	Pz	=	0
+	+	+	+	+	+	+	+				
S7	S6	S5	S4	S3	S2	S1	S0	+	Pz	=	0
=1							=1				



## Reconhecimento do Telegrama

- Mediante a verificação do telegrama, o dispositivo receptor pode verificar a correcta recepção do mesmo e enviar um aviso da confirmação correspondente.
- Se um reconhecimento NAK (recepção incorrecta) for recebido, a transmissão do telegrama é repetida até três vezes.
- Se um reconhecimento BUSY (linha de bus ocupada) for recebido, o participante emissor aguarda um intervalo antes de repetir.

## Reconhecimento do Telegrama

- Se o participante que envia o telegrama não recebe a mensagem de reconhecimento, pára a transmissão.

D7	D6	D5	D4	D3	D2	D1	D0	Direcção de Leitura do Bit de Dados
N	N	0	0	B	B	0	0	Mensagem de Reconhecimento
1	1	0	0	0	0	0	0	BUSY - Ocupado
0	0	0	0	1	1	0	0	NAK - Recepção Incorrecta
1	1	0	0	1	1	0	0	ACK - Recepção Correcta

B = 00 - BUSY

N = 00 - NAK

## Modelo OSI

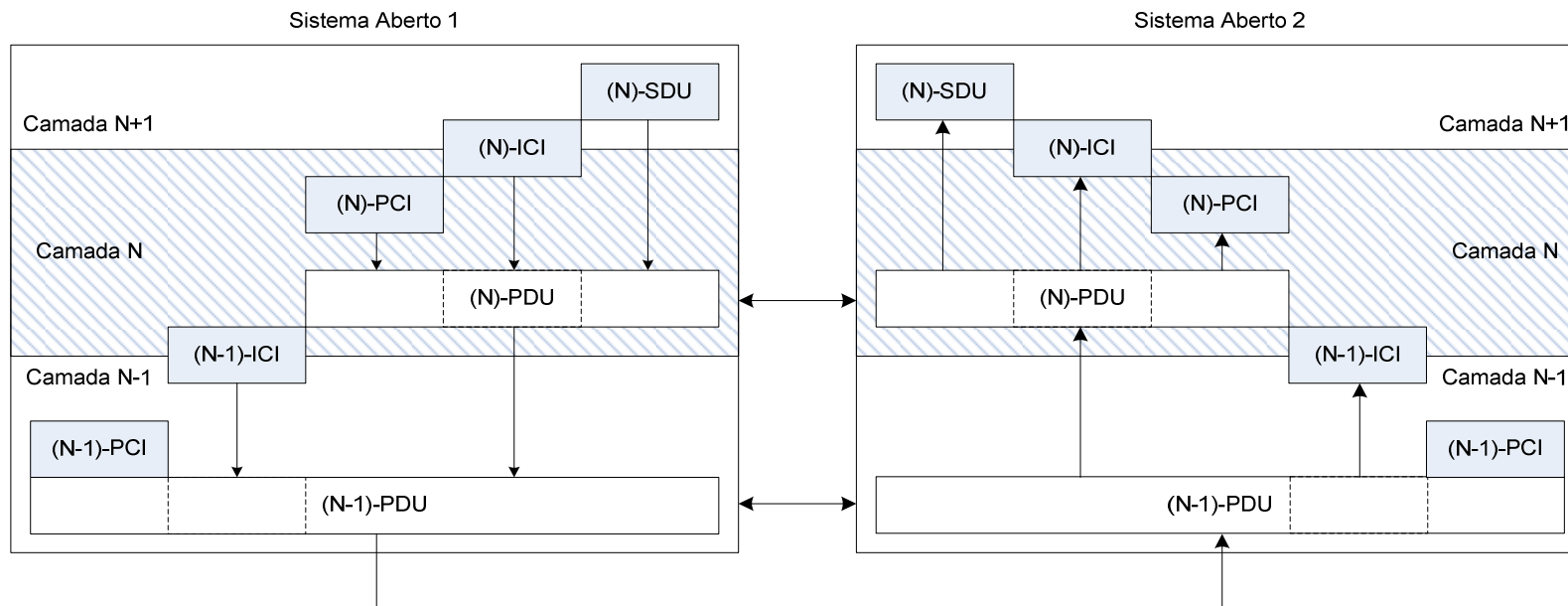
- A base para a descrição funcional do protocolo de comunicação KNX tem como referência o modelo OSI (*Open Systems Interconnection*) concebido nos finais dos anos 70 pela ISO (*International Organization for Standardization*).
- Este modelo garante que sistemas de diferentes fabricantes sejam compatíveis entre si e possam facilmente comunicar.
- A melhor aproximação seria dividir as complexas tarefas da comunicação de dados em sete camadas (*layers*), clarificando as relações entre camadas.

## Modelo OSI

- O modelo OSI veio então implementar 3 importantes conceitos:
  - Serviço (*Service*): Representa qualquer serviço disponível por uma camada à sua camada superior.
  - Interface (*Interface*): Deve existir uma ligação entre duas camadas adjacentes. A interface define que serviços que a camada inferior oferece à camada superior, ou seja, que parâmetros precisam ser transferidos e quais os resultados esperados.
  - Protocolo (*Protocol*): O termo protocolo implica uma série de regras de comunicação entre 2 camadas do mesmo nível.

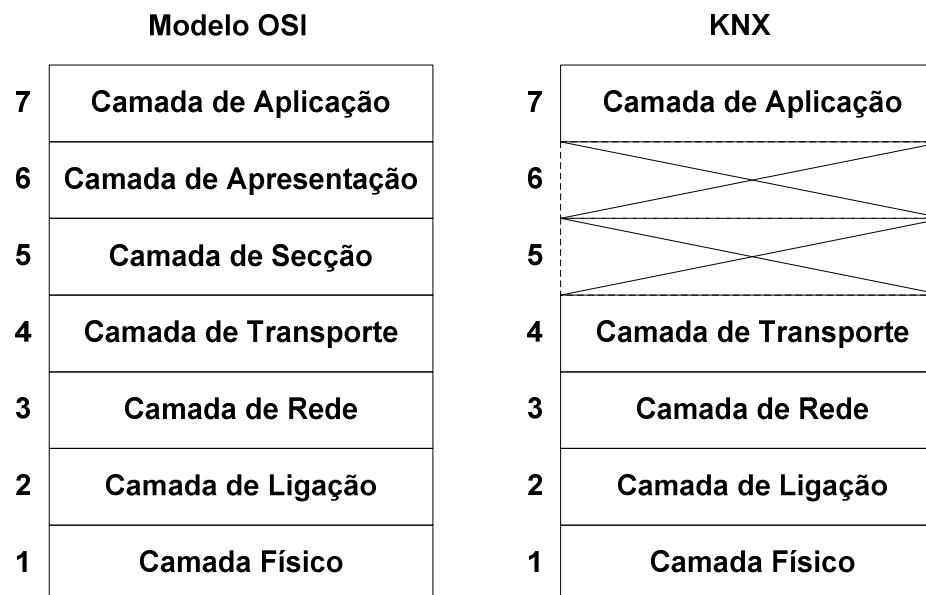
# Modelo OSI

## ➤ Comunicação no Modelo OSI



## Modelo OSI no KNX

- Nem todas as camadas definidas no modelo OSI são necessárias nos sistemas de comunicação. O KNX implementa só 5 das 7 camadas.



## Modelos EIS

- O modelo EIS foi desenvolvido para garantir a compatibilidade entre participantes, mas de fabricantes diferentes.
- O código de informação contém o formato e a estrutura dos objectos de comunicação, tanto para sensores como para actuadores.
- A designação de um modelo EIS é orientada para aplicação para que foi concebida. Isto não significa que o uso de um modelo EIS seja limitado a essa área de aplicação.

## Modelos EIS

- Um código EIS pode também incluir vários objectos de comunicação diferentes.

Tipos de funções KNX

EIS N.º	Função KNX
EIS 1	"Switching"
EIS 2	"Dimming"
EIS 3	"Time"
EIS 4	"Date"
EIS 5	"Value"
EIS 6	"Scaling"
EIS 7	"Drive control"
EIS 8	"Priority"
EIS 9	"Float value"
EIS 10	"16-bit counter value"
EIS 11	"32-bit counter value"