ProgrammerSought ☰

search

[Industrial control old horse] Detailed explanation of Siemens PLC TCP protocol

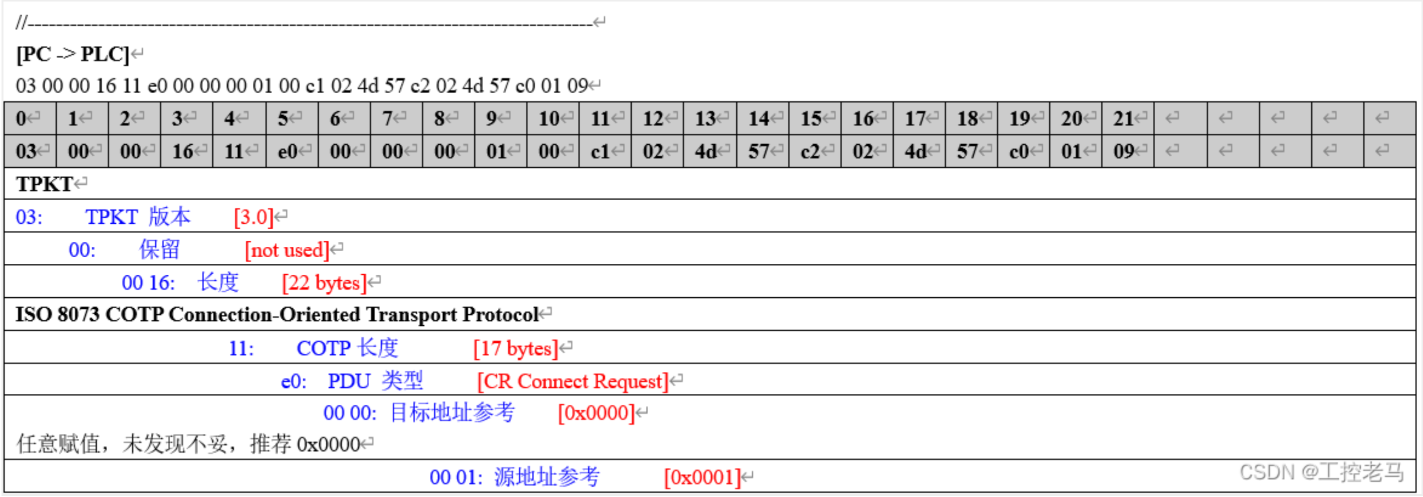
tags: Miscellaneous Notes on Industrial Control Technology Microcontroller embedded hardware manufacture

Detailed explanation of Siemens PLC TCP protocol

Note: Blue text indicates the cracked part, [red text] indicates the numerical description of the cracked part, black text indicates further explanation of the cracked part, black italic bold text indicates the uncracked part, and the highlighted text indicates the driver needs to process For parts that are not highlighted, just keep the default driver processing.

1.Initialize the connection

1.1 S7-200



//--

[PC -> PLC]

03 00 00 16 11 e0 00 00 00 01 00 c1 02 4d 57 c2 02 4d 57 c0 01 09

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21					
03	00	00	16	11	e0	00	00	00	01	00	c1	02	4d	57	c2	02	4d	57	c0	01	09					

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 16: 长度 [22 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

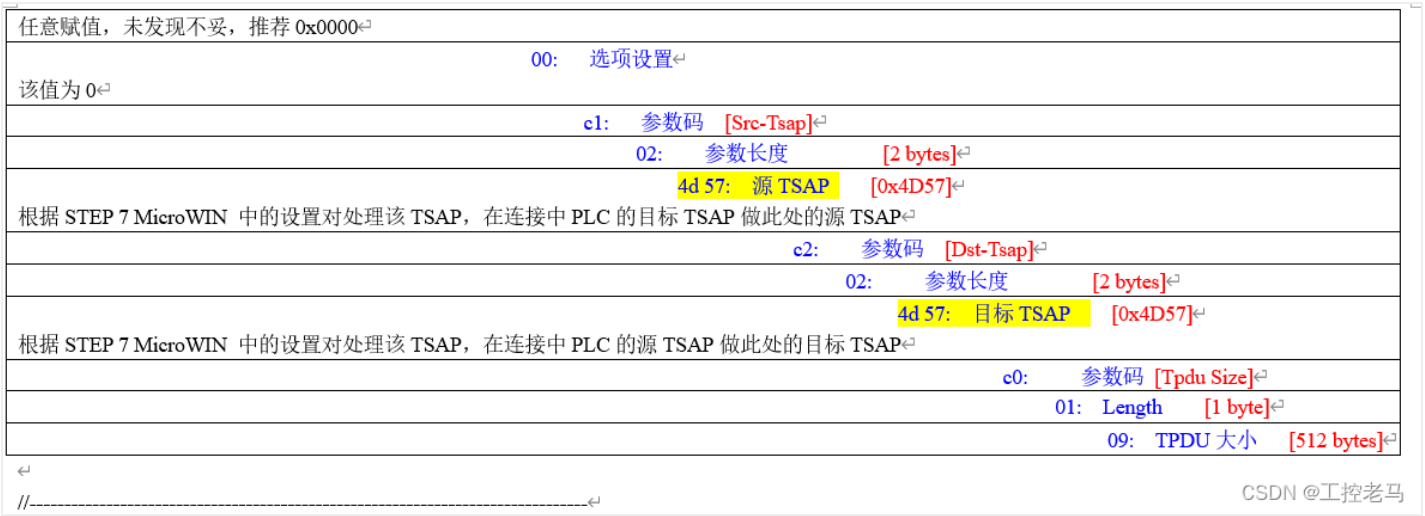
11: COTP 长度 [17 bytes]

e0: PDU 类型 [CR Connect Request]

00 00: 目标地址参考 [0x0000]

任意赋值，未发现不妥，推荐 0x0000

00 01: 源地址参考 [0x0001]



任意赋值，未发现不妥，推荐 0x0000

00: 选项设置

该值为 0

c1: 参数码 [Src-Tsap]

02: 参数长度 [2 bytes]

4d 57: 源 TSAP [0x4D57]

根据 STEP 7 MicroWIN 中的设置对处理该 TSAP，在连接中 PLC 的目标 TSAP 做此处的源 TSAP

c2: 参数码 [Dst-Tsap]

02: 参数长度 [2 bytes]

4d 57: 目标 TSAP [0x4D57]

根据 STEP 7 MicroWIN 中的设置对处理该 TSAP，在连接中 PLC 的源 TSAP 做此处的目标 TSAP

c0: 参数码 [Tpdu Size]

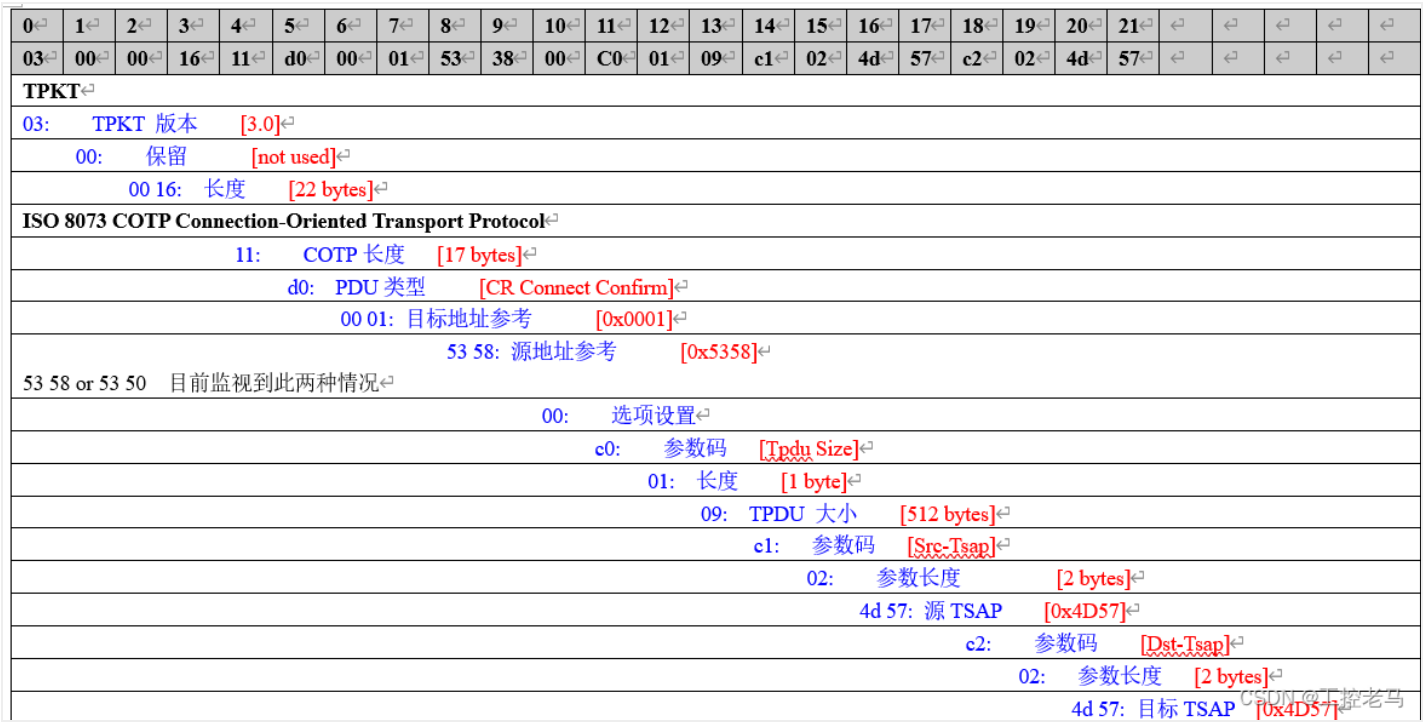
01: Length [1 byte]

09: TPDU 大小 [512 bytes]

//--

[PLC -> PC]

03 00 00 16 11 d0 00 01 53 38 00 c0 01 09 c1 02 4d 57 c2 02 4d 57



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21					
03	00	00	16	11	d0	00	01	53	38	00	C0	01	09	c1	02	4d	57	c2	02	4d	57					

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 16: 长度 [22 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

11: COTP 长度 [17 bytes]

d0: PDU 类型 [CR Connect Confirm]

00 01: 目标地址参考 [0x0001]

53 58: 源地址参考 [0x5358]

53 58 or 53 50 目前监视到此两种情况

00: 选项设置

c0: 参数码 [Tpdu Size]

01: 长度 [1 byte]

09: TPDU 大小 [512 bytes]

c1: 参数码 [Src-Tsap]

02: 参数长度 [2 bytes]

4d 57: 源 TSAP [0x4D57]

c2: 参数码 [Dst-Tsap]

02: 参数长度 [2 bytes]

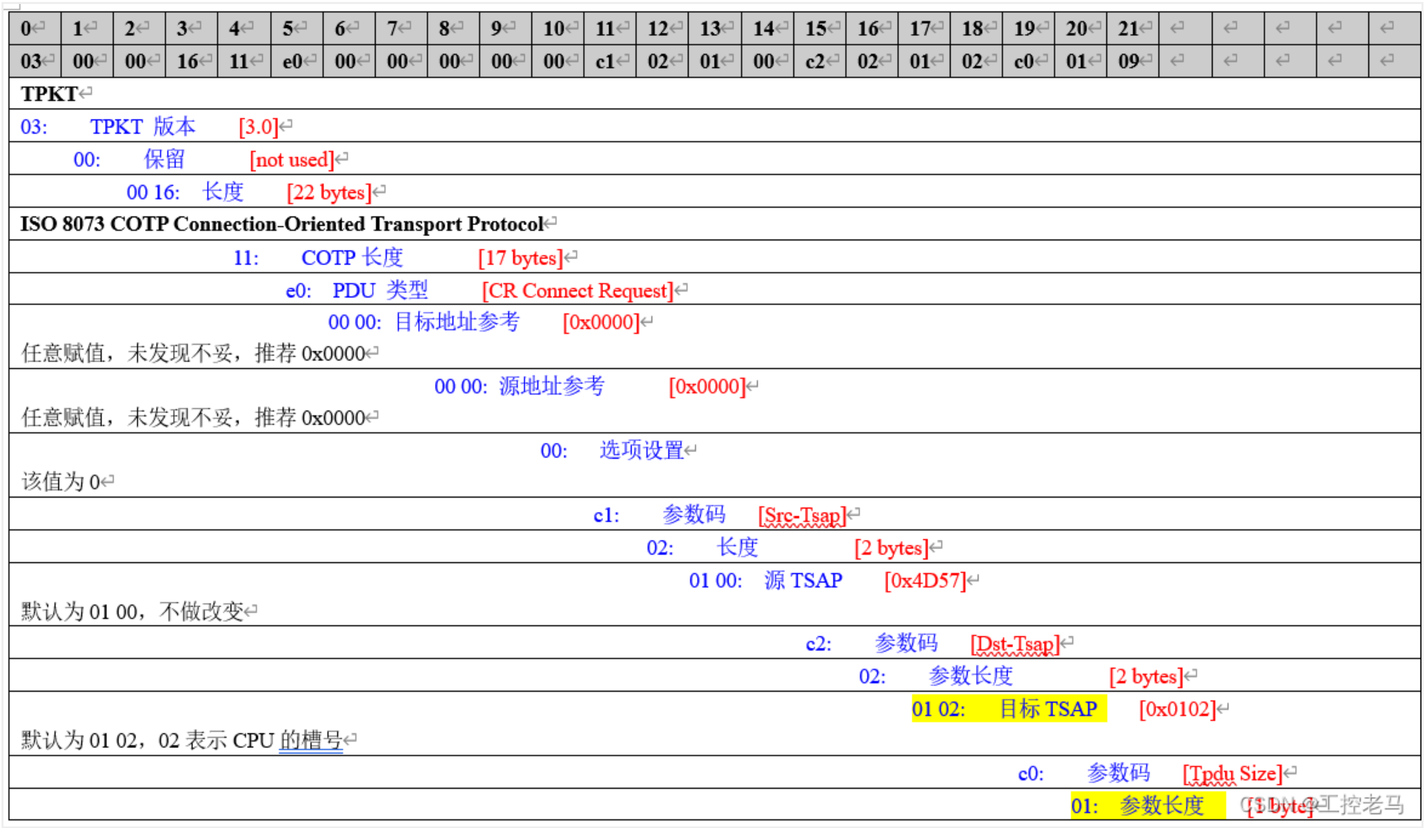
4d 57: 目标 TSAP [0x4D57]

1.2 S7-300

//--

[PC -> PLC]

03 00 00 16 11 e0 00 00 00 00 00 c1 02 01 00 c2 02 01 02 c0 01 09



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21					
03	00	00	16	11	e0	00	00	00	00	00	c1	02	01	00	c2	02	01	02	c0	01	09					

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 16: 长度 [22 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

11: COTP 长度 [17 bytes]

e0: PDU 类型 [CR Connect Request]

00 00: 目标地址参考 [0x0000]

任意赋值，未发现不妥，推荐 0x0000

00 00: 源地址参考 [0x0000]

任意赋值，未发现不妥，推荐 0x0000

00: 选项设置

该值为 0

c1: 参数码 [Src-Tsap]

02: 长度 [2 bytes]

01 00: 源 TSAP [0x4D57]

默认为 01 00，不做改变

c2: 参数码 [Dst-Tsap]

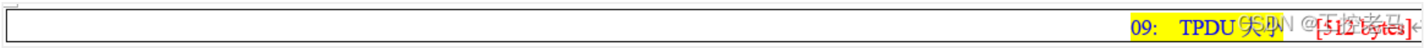
02: 参数长度 [2 bytes]

01 02: 目标 TSAP [0x0102]

默认为 01 02，02 表示 CPU 的槽号

c0: 参数码 [Tpdu Size]

01: 参数长度 [1 byte]

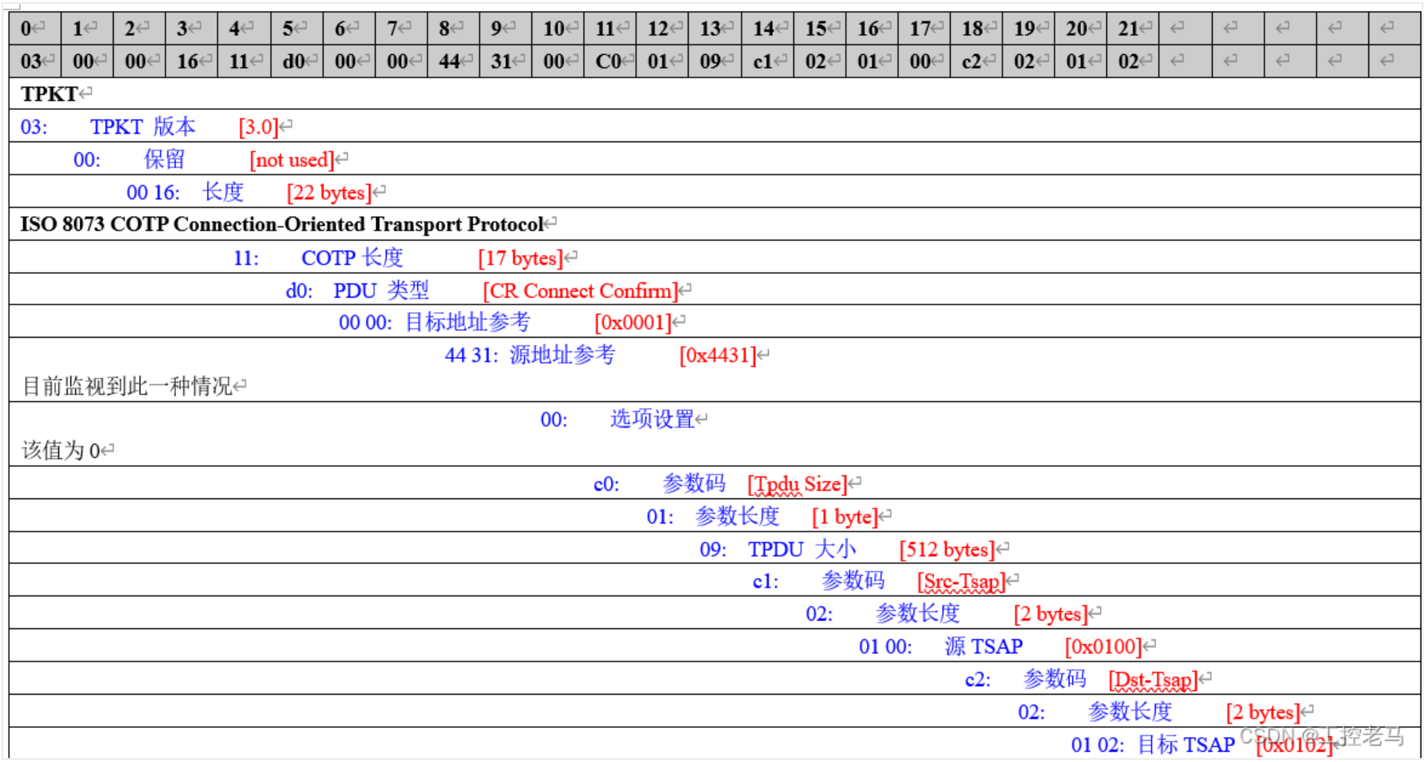


09: TPDU 大小 [512 bytes]

//---

[PLC -> PC]

03 00 00 16 11 d0 00 00 44 31 00 c0 01 09 c1 02 01 00 c2 02 01 02



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21					
03	00	00	16	11	d0	00	00	44	31	00	C0	01	09	c1	02	01	00	c2	02	01	02					

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 16: 长度 [22 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

11: COTP 长度 [17 bytes]

d0: PDU 类型 [CR Connect Confirm]

00 00: 目标地址参考 [0x0001]

44 31: 源地址参考 [0x4431]

目前监视到此一种情况

00: 选项设置

该值为 0

c0: 参数码 [Tpdu Size]

01: 参数长度 [1 byte]

09: TPDU 大小 [512 bytes]

c1: 参数码 [Src-Tsap]

02: 参数长度 [2 bytes]

01 00: 源 TSAP [0x0100]

c2: 参数码 [Dst-Tsap]

02: 参数长度 [2 bytes]

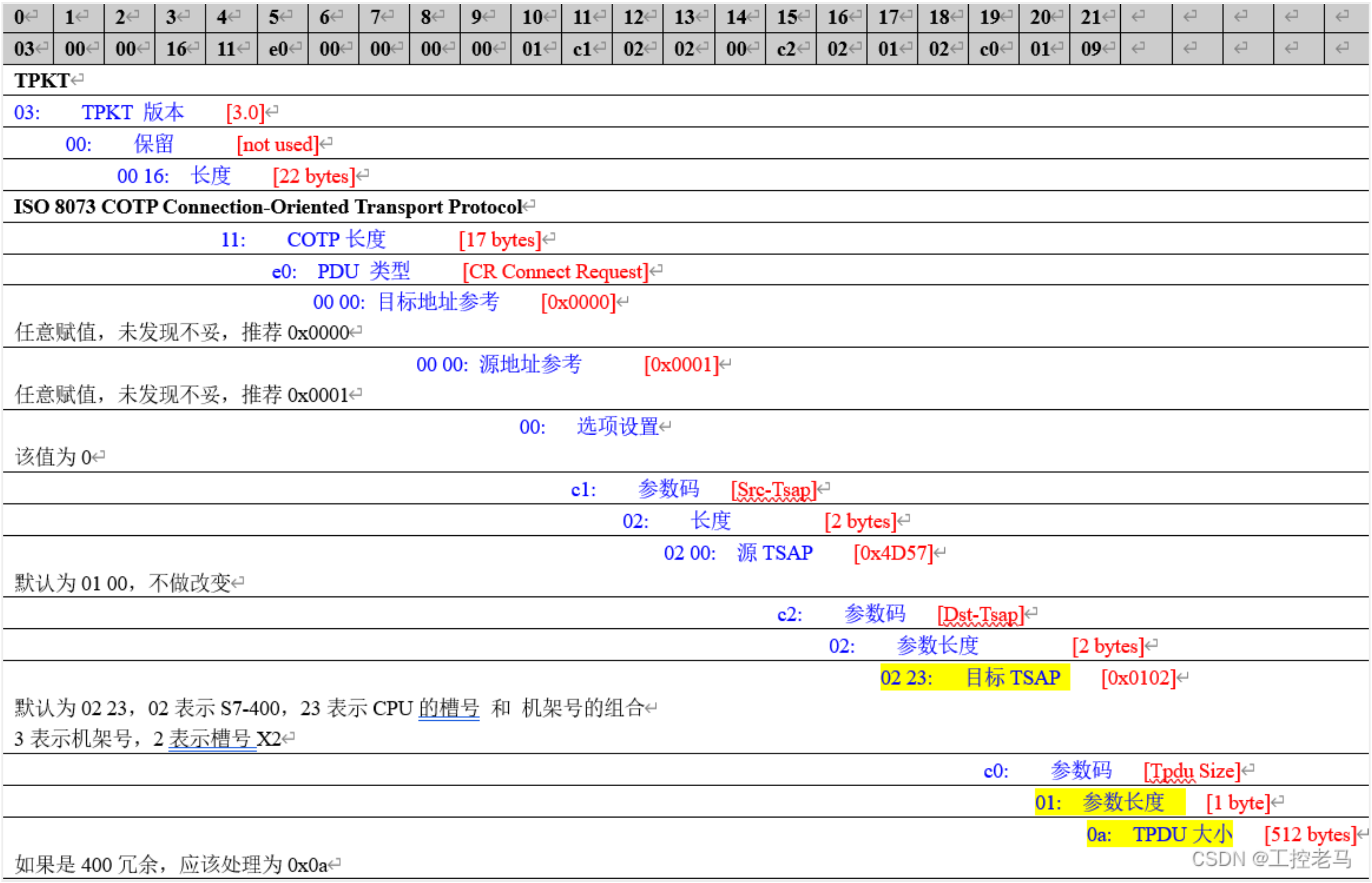
01 02: 目标 TSAP [0x0102]

1.3 S7-400

//--

[PC -> PLC]

03 00 00 16 11 e0 00 00 00 01 00 c1 02 02 00 c2 02 02 23 c0 01 09



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21					
03	00	00	16	11	e0	00	00	00	00	01	c1	02	02	00	c2	02	01	02	c0	01	09					

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 16: 长度 [22 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

11: COTP 长度 [17 bytes]

e0: PDU 类型 [CR Connect Request]

00 00: 目标地址参考 [0x0000]

任意赋值，未发现不妥，推荐 0x0000

00 00: 源地址参考 [0x0001]

任意赋值，未发现不妥，推荐 0x0001

00: 选项设置

该值为 0

c1: 参数码 [Src-Tsap]

02: 长度 [2 bytes]

02 00: 源 TSAP [0x4D57]

默认为 01 00，不做改变

c2: 参数码 [Dst-Tsap]

02: 参数长度 [2 bytes]

02 23: 目标 TSAP [0x0102]

默认为 02 23，02 表示 S7-400，23 表示 CPU 的槽号 和 机架号的组合
3 表示机架号，2 表示槽号 X2

c0: 参数码 [Tpdu Size]

01: 参数长度 [1 byte]

0a: TPDU 大小 [512 bytes]

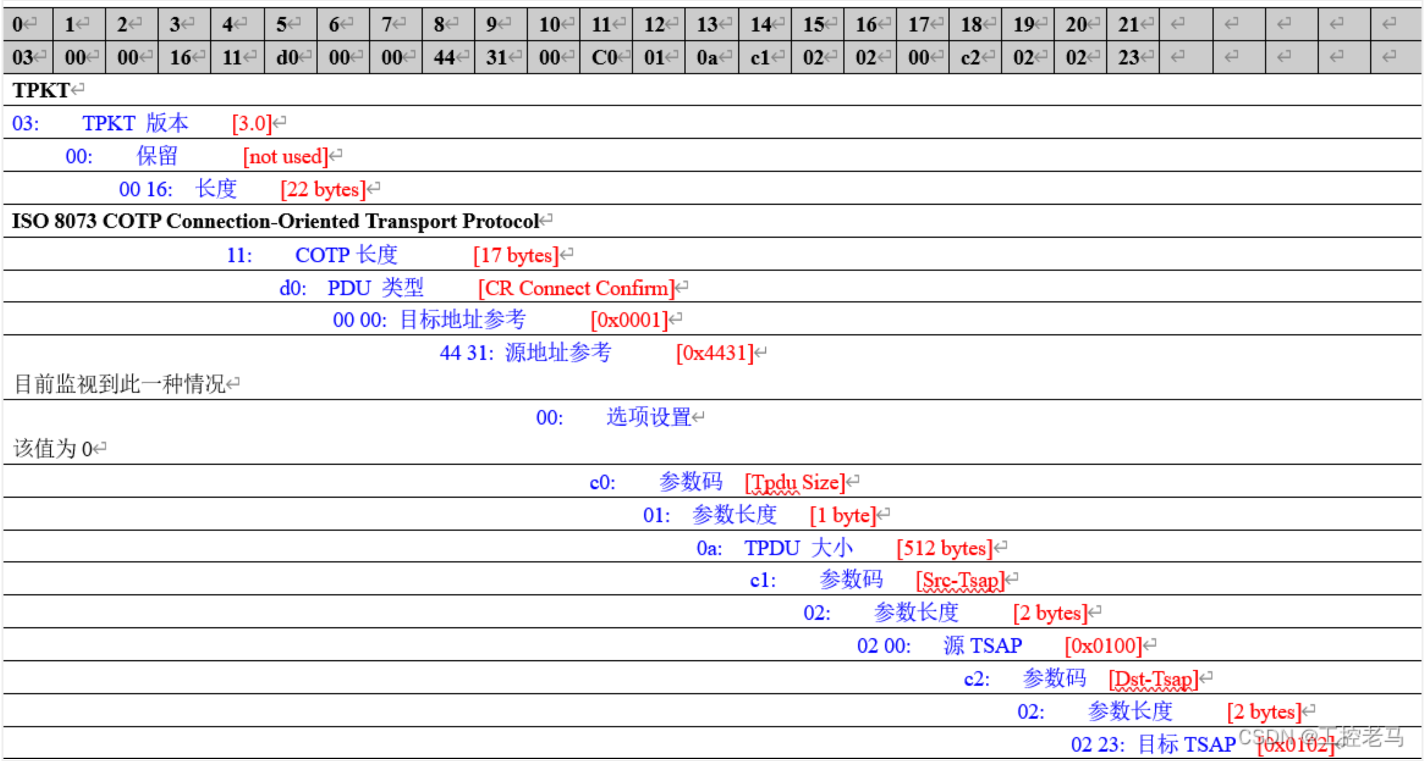
如果是 400 冗余，应该处理为 0x0a

CSDN @工控老马

//--

[PLC -> PC]

03 00 00 16 11 d0 00 00 44 31 00 c0 01 0ac1 02 01 00 c2 02 01 02



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21					
03	00	00	16	11	d0	00	00	44	31	00	C0	01	0a	c1	02	02	00	c2	02	02	23					

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 16: 长度 [22 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

11: COTP 长度 [17 bytes]

d0: PDU 类型 [CR Connect Confirm]

00 00: 目标地址参考 [0x0001]

44 31: 源地址参考 [0x4431]
目前监视到此一种情况

00: 选项设置
该值为 0

c0: 参数码 [Tpdu Size]

01: 参数长度 [1 byte]

0a: TPDU 大小 [512 bytes]

c1: 参数码 [Src-Tsap]

02: 参数长度 [2 bytes]

02 00: 源 TSAP [0x0100]

c2: 参数码 [Dst-Tsap]

02: 参数长度 [2 bytes]

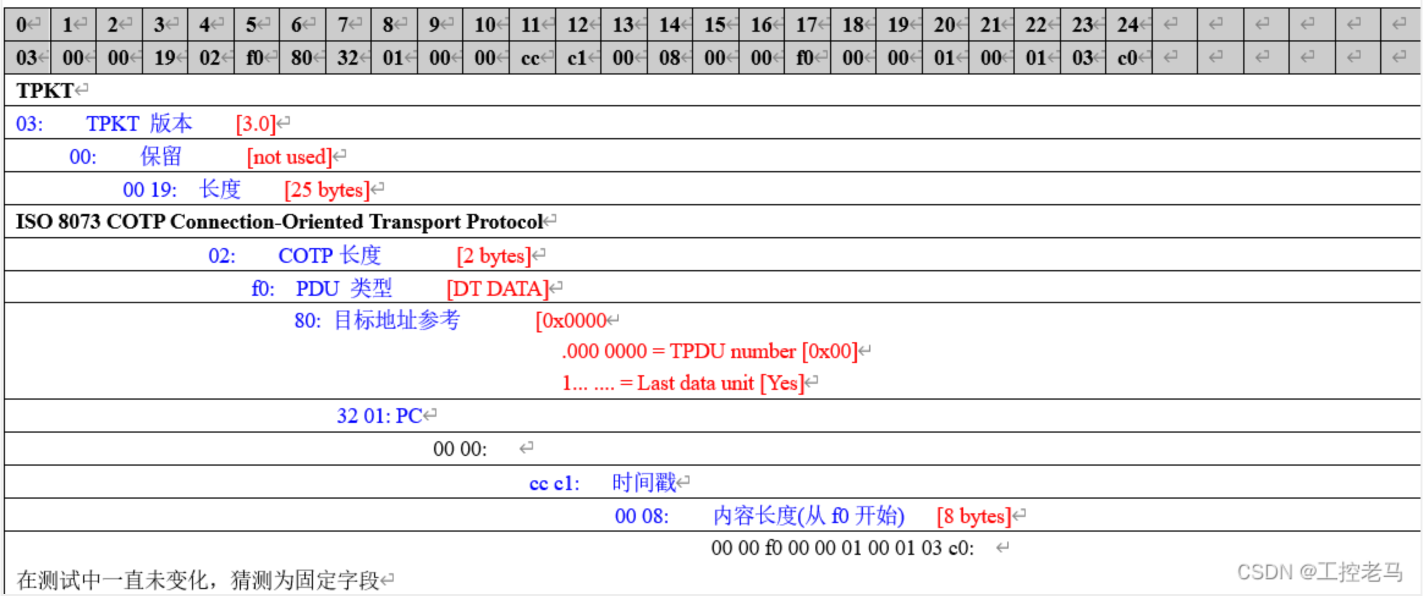
02 23: 目标 TSAP [0x0102]

2 Initialize communication

//---

[PC -> PLC]

03 00 00 19 02 f0 80 32 01 00 00 cc c1 00 08 00 00 f0 00 00 01 00 01 03 c0



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24						
03	00	00	19	02	f0	80	32	01	00	00	cc	c1	00	08	00	00	f0	00	00	01	00	01	03	c0						

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 19: 长度 [25 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

02: COTP 长度 [2 bytes]

f0: PDU 类型 [DT DATA]

80: 目标地址参考 [0x0000
.000 0000 = TPDU number [0x00]
1... .... = Last data unit [Yes]

32 01: PC

00 00:

cc c1: 时间戳

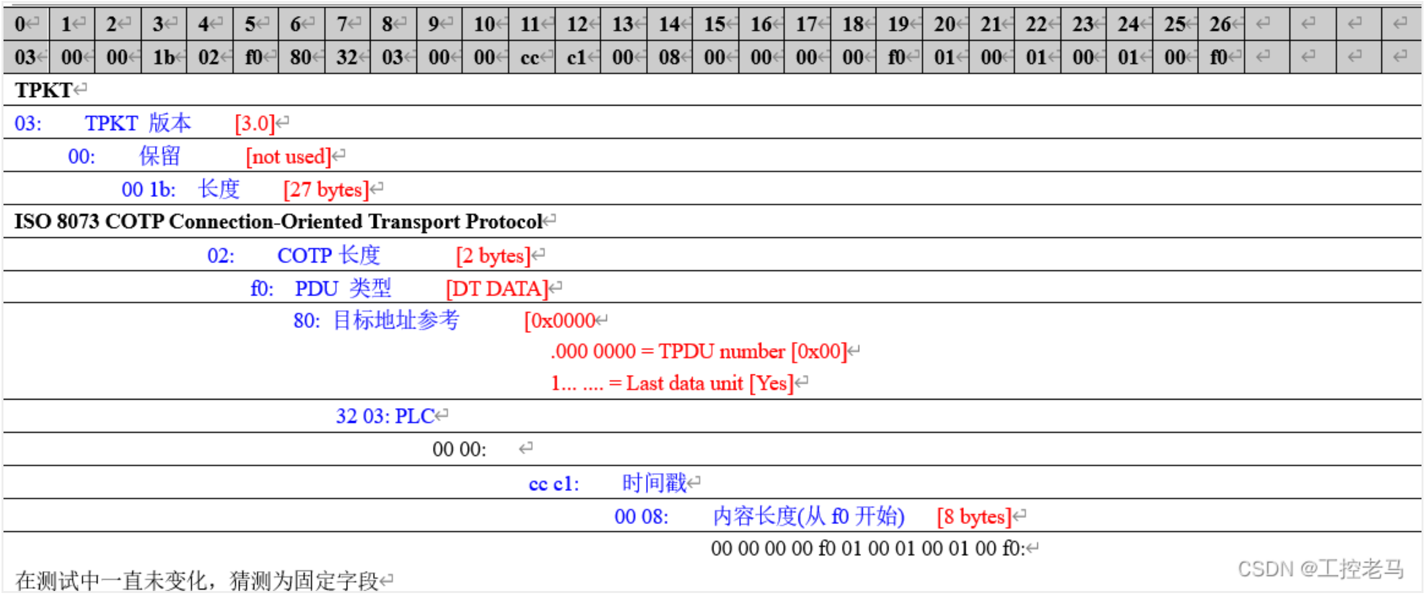
00 08: 内容长度(从 f0 开始) [8 bytes]

00 00 f0 00 00 01 00 01 03 c0:
在测试中一直未变化，猜测为固定字段

//--

[PLC -> PC]

03 00 00 1b 02 f0 80 32 03 00 00 cc c1 00 08 00 00 00 00 f0 01 00 01 00 01 00 f0



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26				
03	00	00	1b	02	f0	80	32	03	00	00	cc	c1	00	08	00	00	00	00	f0	01	00	01	00	01	00	f0				

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 1b: 长度 [27 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

02: COTP 长度 [2 bytes]

f0: PDU 类型 [DT DATA]

80: 目标地址参考 [0x0000

.000 0000 = TPDU number [0x00]

1... .... = Last data unit [Yes]

32 03: PLC

00 00:

cc c1: 时间戳

00 08: 内容长度(从 f0 开始) [8 bytes]

00 00 00 00 f0 01 00 01 00 01 00 f0:

在测试中一直未变化，猜测为固定字段

CSDN @工控老马

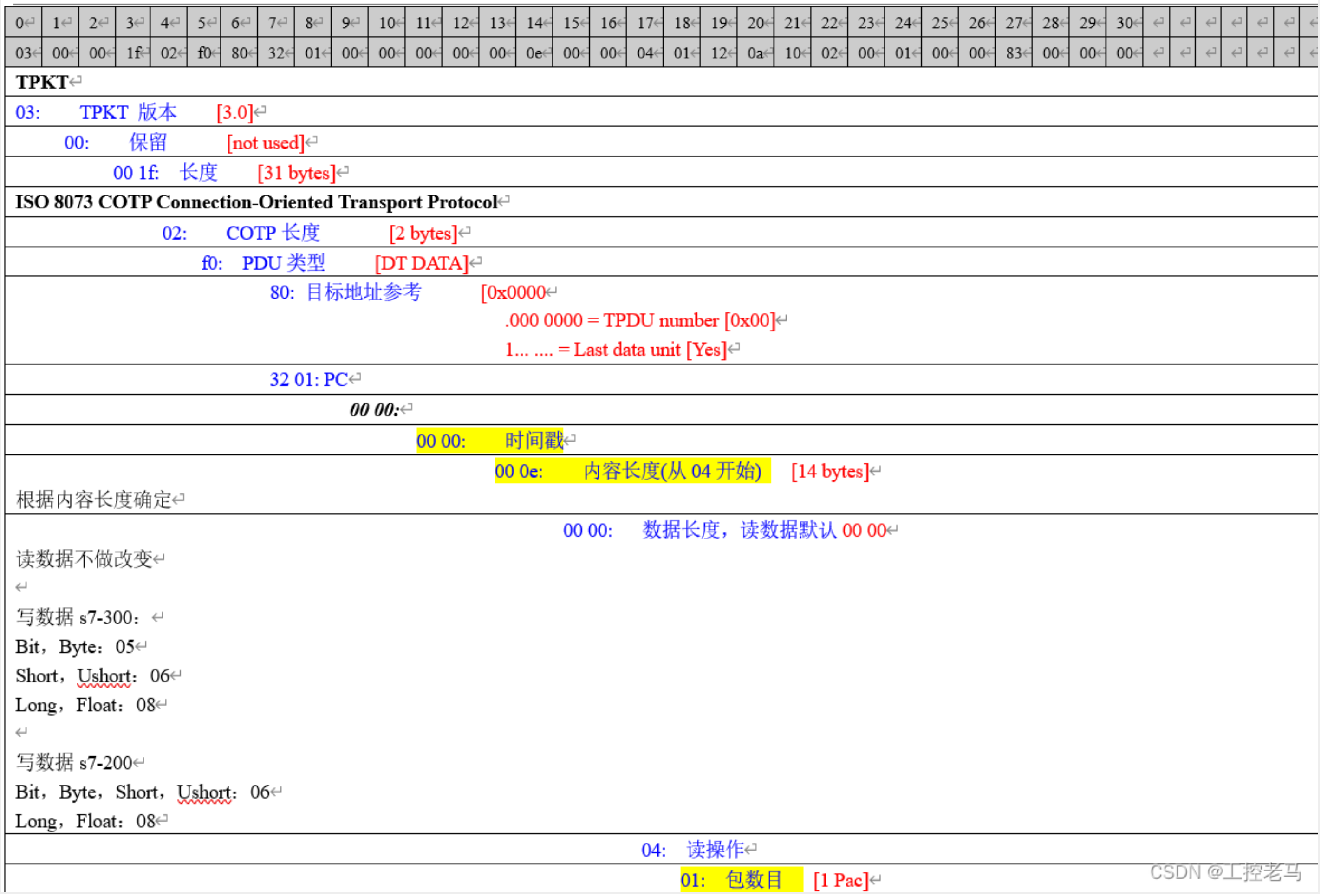
3. Read data

3.1 Typical example [M0]

//--

[PC -> PLC]

03 00 00 1f 02 f0 80 32 01 00 00 00 00 00 0e 00 00 04 01 12 0a 10 02 00 01 00 00 83 00 00 00



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30							
03	00	00	1f	02	f0	80	32	01	00	00	00	00	00	0e	00	00	04	01	12	0a	10	02	00	01	00	00	83	00	00	00							

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 1f: 长度 [31 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

02: COTP 长度 [2 bytes]

f0: PDU 类型 [DT DATA]

80: 目标地址参考 [0x0000
.000 0000 = TPDU number [0x00]
1... .... = Last data unit [Yes]

32 01: PC

00 00:

00 00: 时间戳

00 0e: 内容长度(从 04 开始) [14 bytes]
根据内容长度确定

00 00: 数据长度，读数据默认 00 00
读数据不做改变

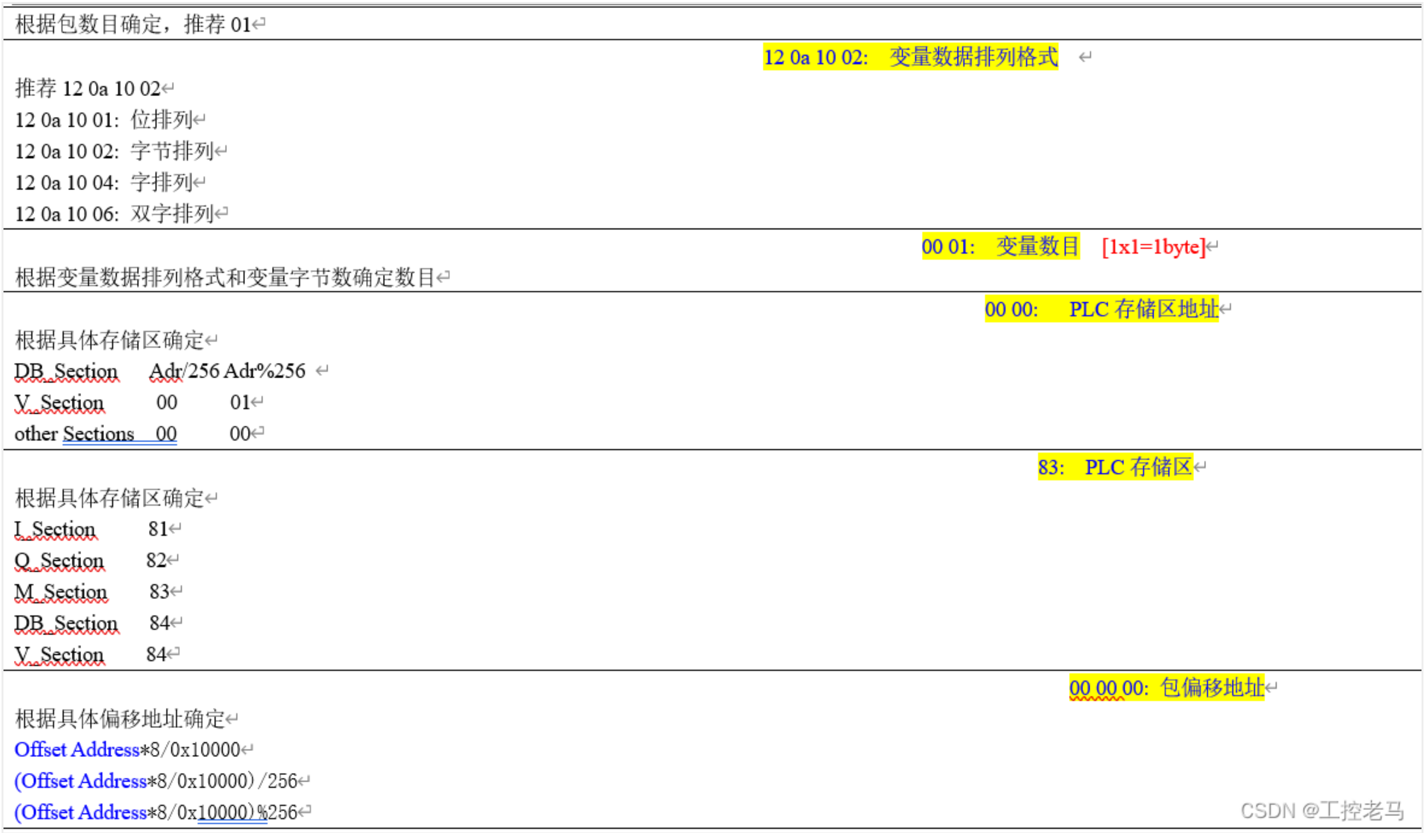
写数据 s7-300：
Bit，Byte：05
Short，Ushort：06
Long，Float：08

写数据 s7-200
Bit，Byte，Short，Ushort：06
Long，Float：08

04: 读操作

01: 包数目 [1 Pac]

CSDN @工控老马



根据包数目确定，推荐 01

12 0a 10 02: 变量数据排列格式
推荐 12 0a 10 02
12 0a 10 01：位排列
12 0a 10 02：字节排列
12 0a 10 04：字排列
12 0a 10 06：双字排列

00 01: 变量数目 [1x1=1byte]
根据变量数据排列格式和变量字节数确定数目

00 00: PLC 存储区地址
根据具体存储区确定

DB Section	Adr/256	Adr%256
V Section	00	01
other Sections	00	00

83: PLC 存储区
根据具体存储区确定

I Section	81
Q Section	82
M Section	83
DB Section	84
V Section	84

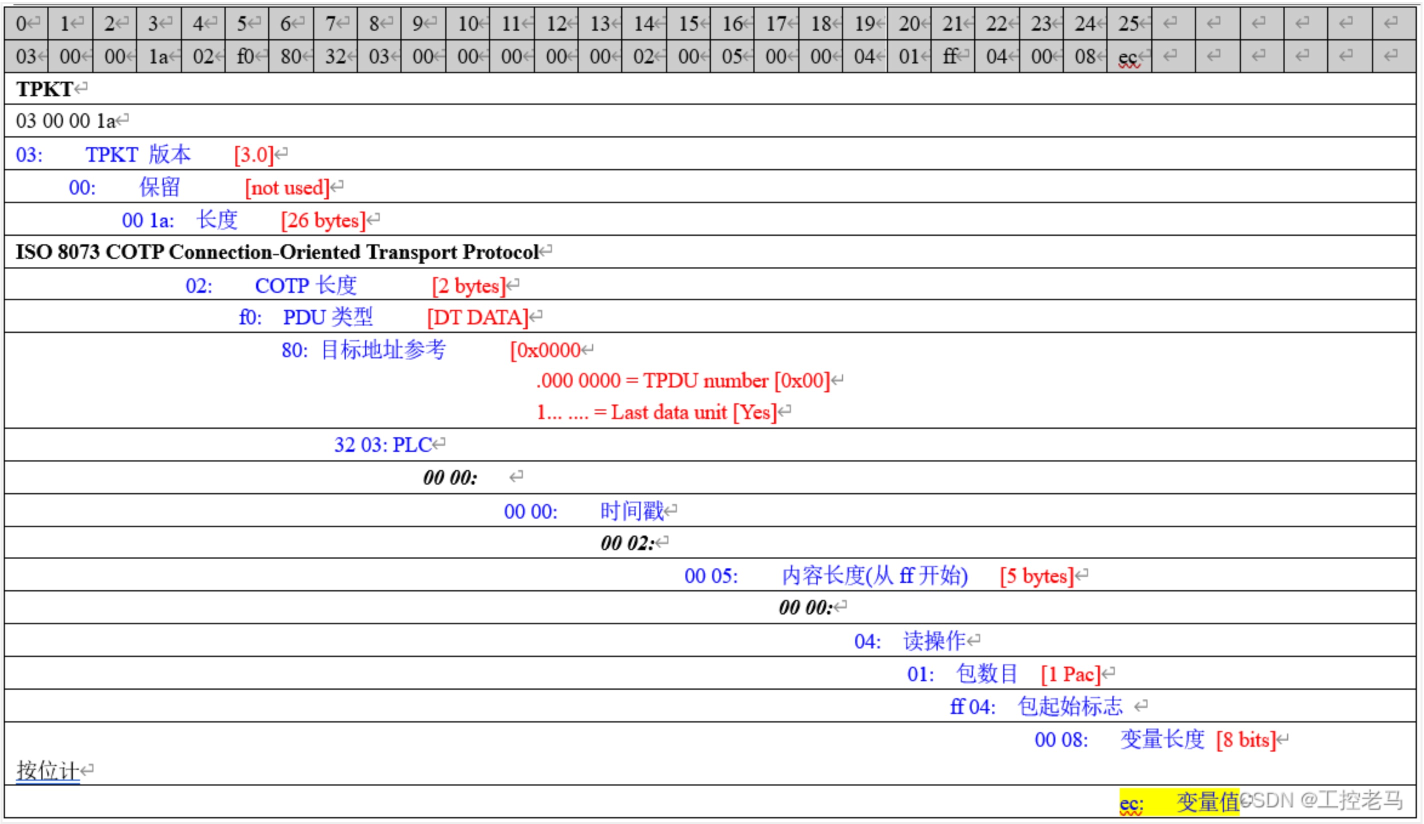
00 00 00: 包偏移地址
根据具体偏移地址确定
Offset Address*8/0x10000
(Offset Address*8/0x10000)/256
(Offset Address*8/0x10000)%256

CSDN @工控老马

//---

[PLC -> PC]

03 00 00 1a 02 f0 80 32 03 00 00 00 00 00 02 00 05 00 00 04 01 ff 04 00 08 ec



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
03	00	00	1a	02	f0	80	32	03	00	00	00	00	00	02	00	05	00	00	04	01	ff	04	00	08	ec

TPKT

03 00 00 1a

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 1a: 长度 [26 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

02: COTP 长度 [2 bytes]

f0: PDU 类型 [DT DATA]

80: 目标地址参考 [0x0000
.000 0000 = TPDU number [0x00]
1... .... = Last data unit [Yes]

32 03: PLC

00 00:

00 00: 时间戳

00 02:

00 05: 内容长度(从 ff 开始) [5 bytes]

00 00:

04: 读操作

01: 包数目 [1 Pac]

ff 04: 包起始标志

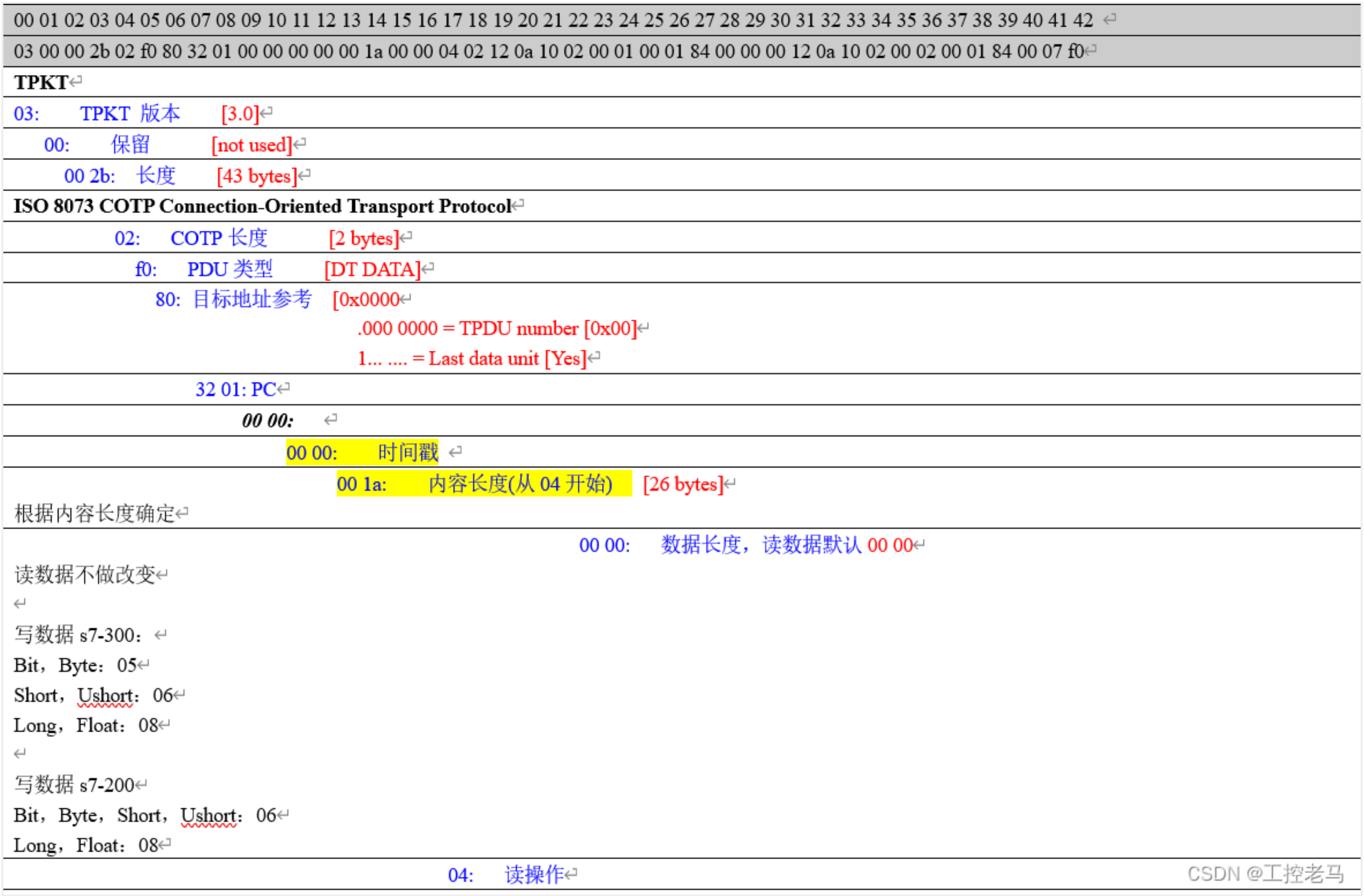
00 08: 变量长度 [8 bits]
按位计

ec: 变量值

3.2 Reference example [VB0 VB254 VB255]

//---

[PC -> PLC]



```
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
03 00 00 2b 02 f0 80 32 01 00 00 00 00 00 1a 00 00 04 02 12 0a 10 02 00 01 00 01 84 00 00 00 12 0a 10 02 00 02 00 01 84 00 07 f0
```

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 2b: 长度 [43 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

02: COTP 长度 [2 bytes]

f0: PDU 类型 [DT DATA]

80: 目标地址参考 [0x0000

.000 0000 = TPDU number [0x00]

1... .... = Last data unit [Yes]

32 01: PC

00 00:

00 00: 时间戳

00 1a: 内容长度(从 04 开始) [26 bytes]

根据内容长度确定

00 00: 数据长度，读数据默认 00 00

读数据不做改变

写数据 s7-300：

Bit，Byte：05

Short，Ushort：06

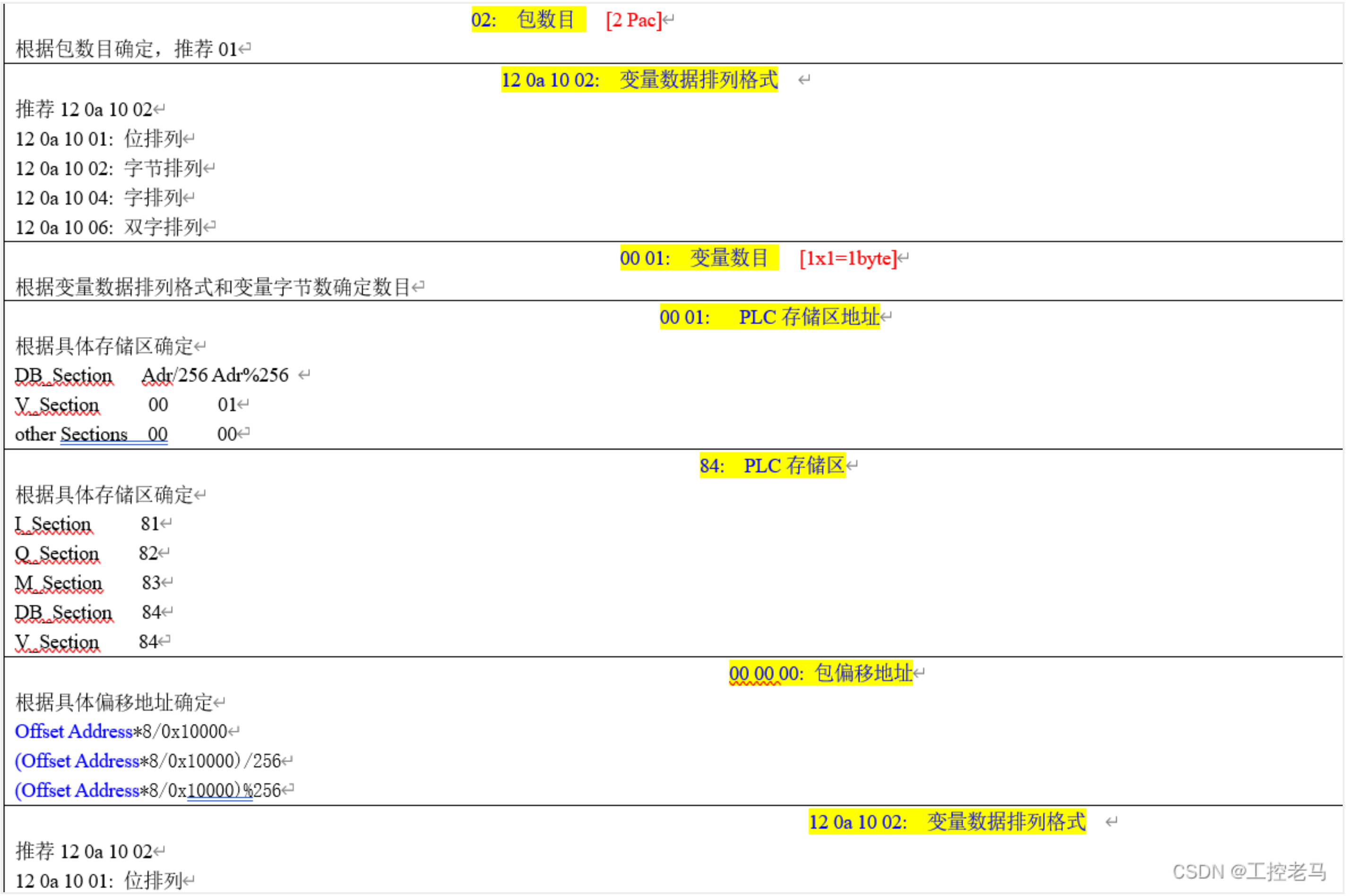
Long，Float：08

写数据 s7-200

Bit，Byte，Short，Ushort：06

Long，Float：08

04: 读操作



02: 包数目 [2 Pac]

根据包数目确定，推荐 01

12 0a 10 02: 变量数据排列格式

推荐 12 0a 10 02

12 0a 10 01: 位排列

12 0a 10 02: 字节排列

12 0a 10 04: 字排列

12 0a 10 06: 双字排列

00 01: 变量数目 [1x1=1byte]

根据变量数据排列格式和变量字节数确定数目

00 01: PLC 存储区地址

根据具体存储区确定

DB_Section	Adr/256	Adr%256
V_Section	00	01
other Sections	00	00

84: PLC 存储区

根据具体存储区确定

I_Section	81
Q_Section	82
M_Section	83
DB_Section	84
V_Section	84

00 00 00: 包偏移地址

根据具体偏移地址确定

Offset Address*8/0x10000

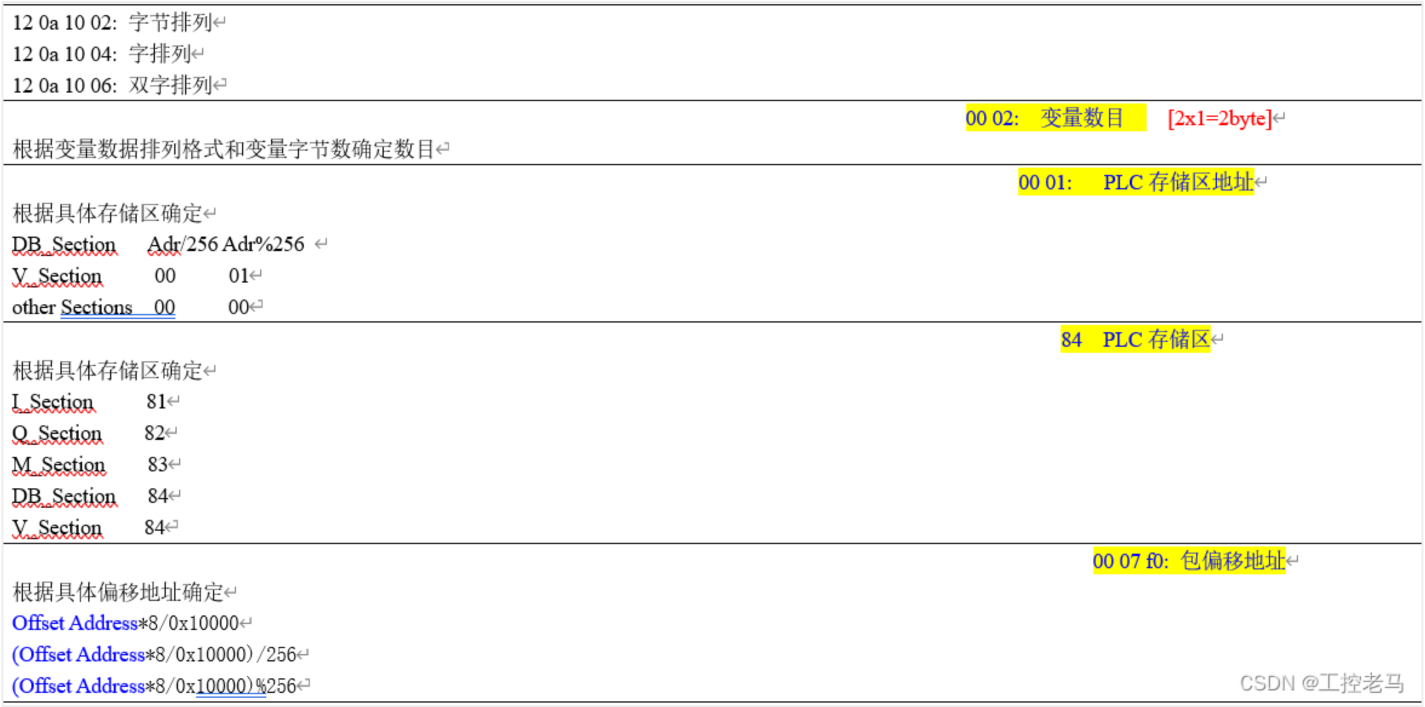
(Offset Address*8/0x10000)/256

(Offset Address*8/0x10000)%256

12 0a 10 02: 变量数据排列格式

推荐 12 0a 10 02

12 0a 10 01: 位排列



12 0a 10 02: 字节排列
12 0a 10 04: 字排列
12 0a 10 06: 双字排列

00 02: 变量数目 [2x1=2byte]
根据变量数据排列格式和变量字节数确定数目

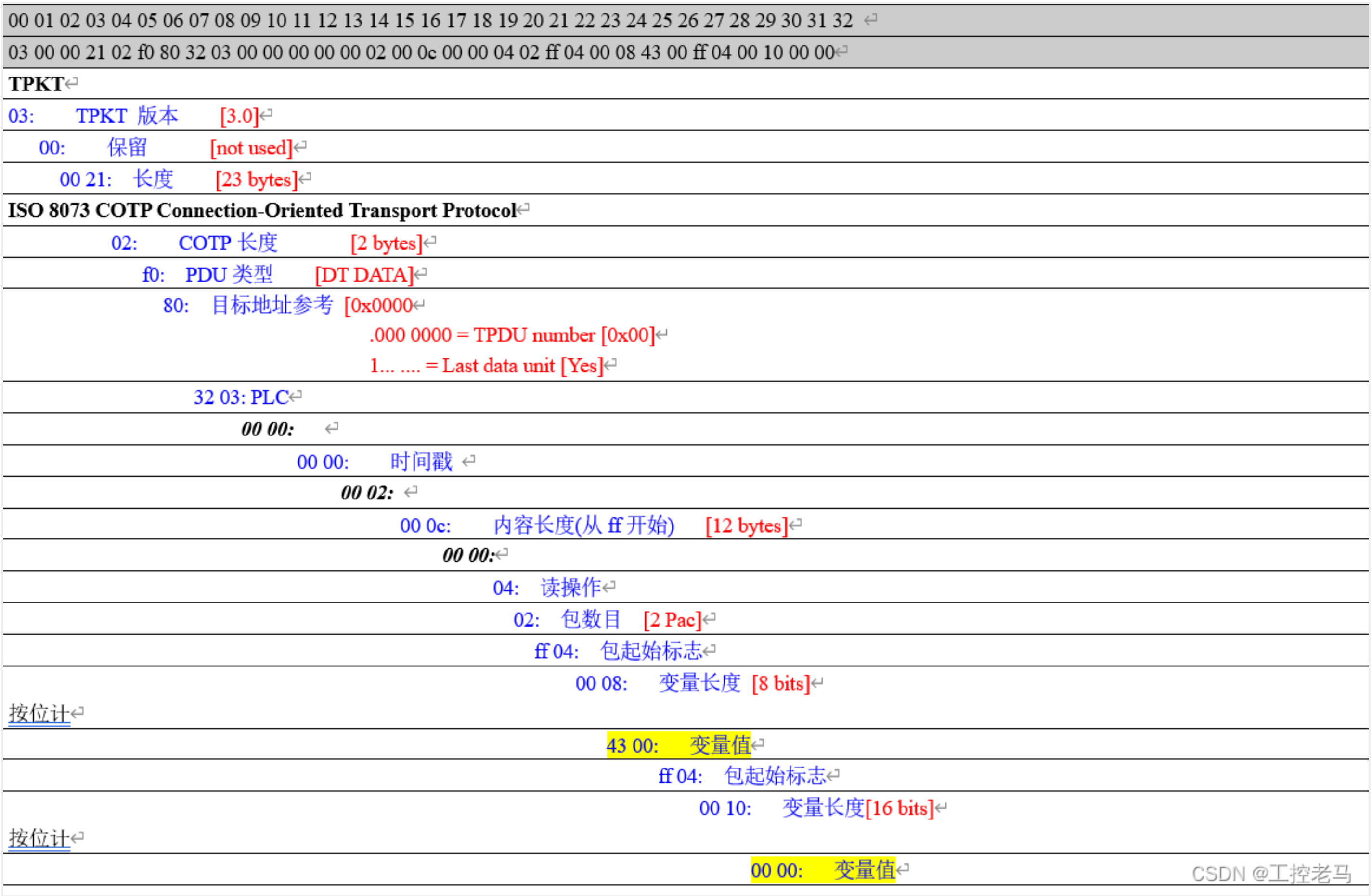
00 01: PLC 存储区地址
根据具体存储区确定
DB Section Adr/256 Adr%256
V Section 00 01
other Sections 00 00

84 PLC 存储区
根据具体存储区确定
I Section 81
Q Section 82
M Section 83
DB Section 84
V Section 84

00 07 f0: 包偏移地址
根据具体偏移地址确定
Offset Address*8/0x10000
(Offset Address*8/0x10000)/256
(Offset Address*8/0x10000)%256

//---

[PLC -> PC]



00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
03 00 00 21 02 f0 80 32 03 00 00 00 00 00 02 00 0c 00 00 04 02 ff 04 00 08 43 00 ff 04 00 10 00 00
TPKT
03: TPKT 版本 [3.0]
00: 保留 [not used]
00 21: 长度 [23 bytes]
ISO 8073 COTP Connection-Oriented Transport Protocol
02: COTP 长度 [2 bytes]
f0: PDU 类型 [DT DATA]
80: 目标地址参考 [0x0000 .000 0000 = TPDU number [0x00] 1... .... = Last data unit [Yes]
32 03: PLC
00 00:
00 00: 时间戳
00 02:
00 0c: 内容长度(从 ff 开始) [12 bytes]
00 00:
04: 读操作
02: 包数目 [2 Pac]
ff 04: 包起始标志
00 08: 变量长度 [8 bits] 按位计
43 00: 变量值
ff 04: 包起始标志
00 10: 变量长度[16 bits] 按位计
00 00: 变量值

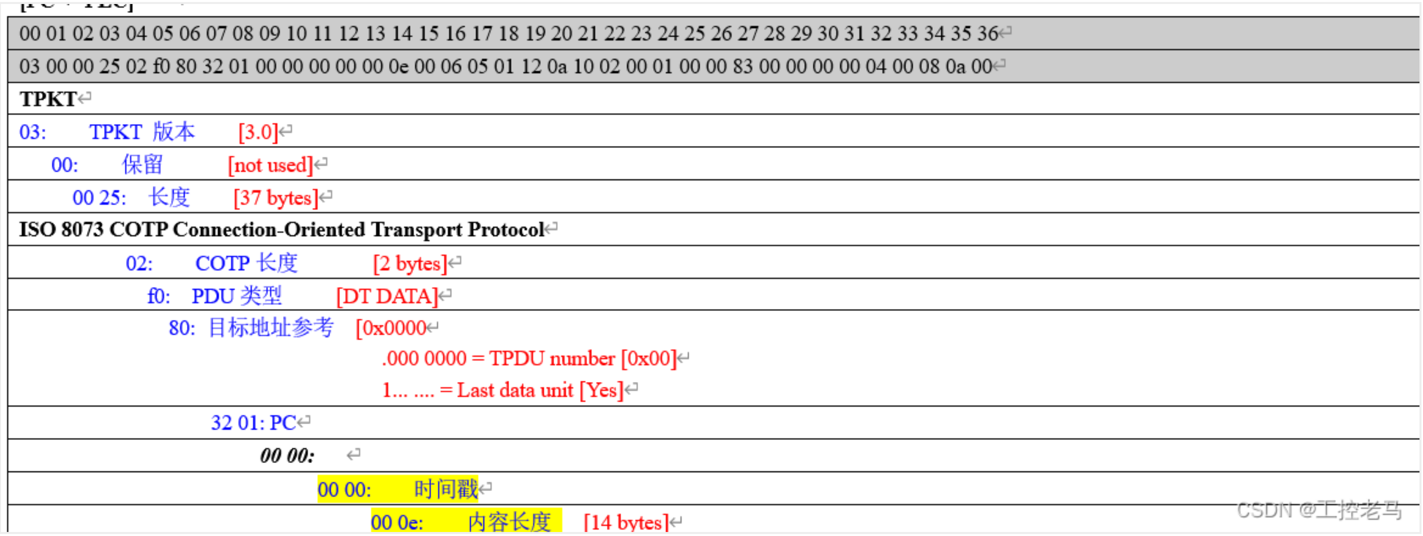
4 Write data

4.1 S7-200

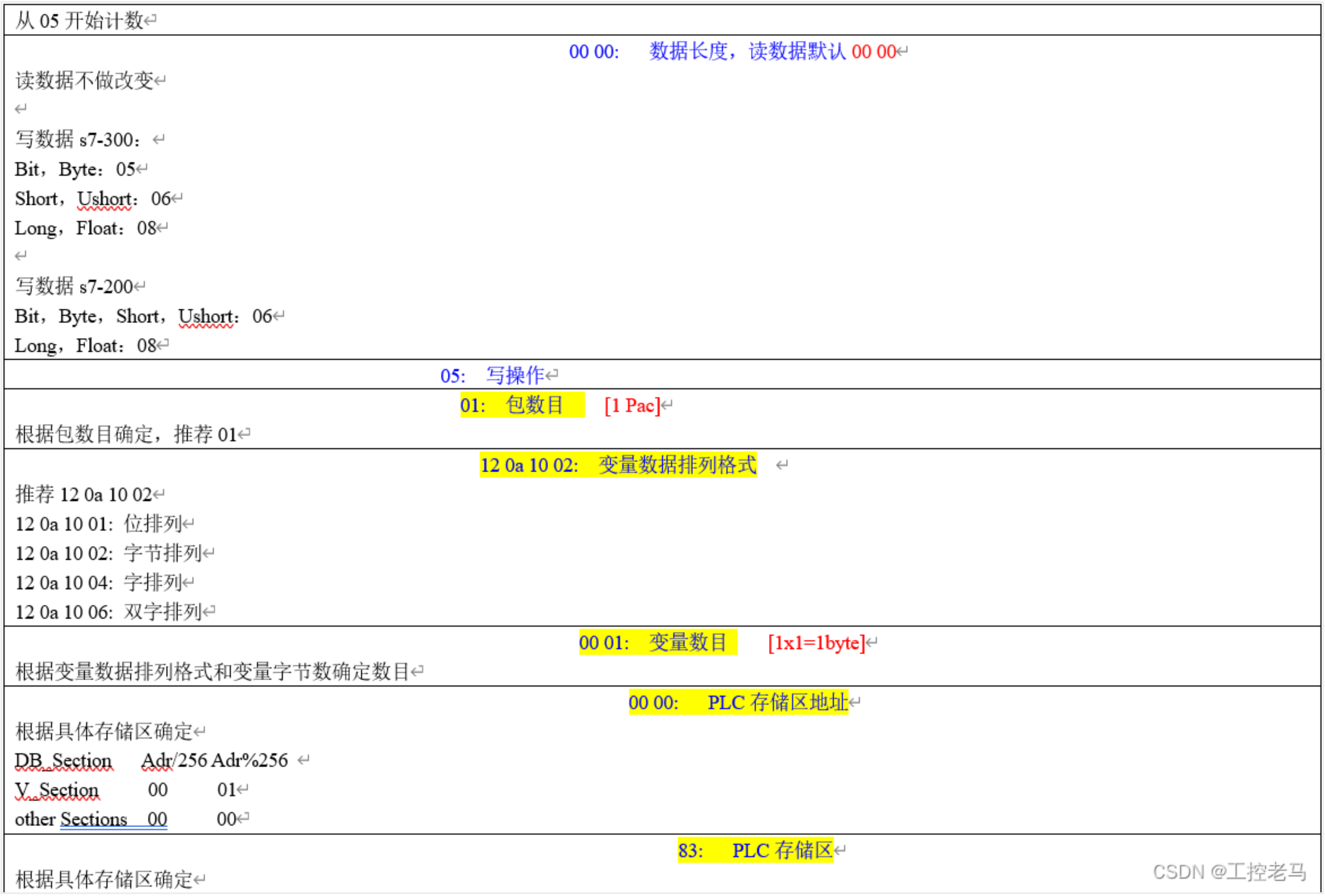
4.1.1 Typical example [MB0]

//---

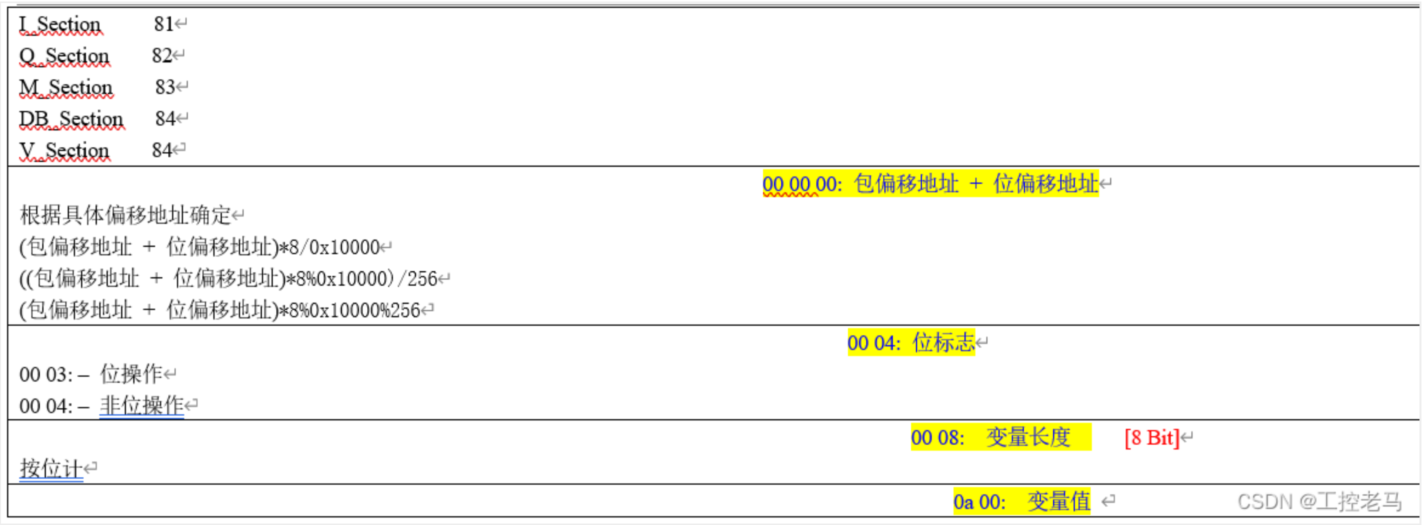
[PC -> PLC]



00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
03 00 00 25 02 f0 80 32 01 00 00 00 00 00 0e 00 06 05 01 12 0a 10 02 00 01 00 00 83 00 00 00 00 04 00 08 0a 00
TPKT
03: TPKT 版本 [3.0]
00: 保留 [not used]
00 25: 长度 [37 bytes]
ISO 8073 COTP Connection-Oriented Transport Protocol
02: COTP 长度 [2 bytes]
f0: PDU 类型 [DT DATA]
80: 目标地址参考 [0x0000 .000 0000 = TPDU number [0x00] 1... .... = Last data unit [Yes]
32 01: PC
00 00:
00 00: 时间戳
00 0e: 内容长度 [14 bytes]



从 05 开始计数
00 00: 数据长度，读数据默认 00 00 读数据不做改变 写数据 s7-300： Bit，Byte：05 Short，Ushort：06 Long，Float：08 写数据 s7-200 Bit，Byte，Short，Ushort：06 Long，Float：08
05: 写操作
01: 包数目 [1 Pac] 根据包数目确定，推荐 01
12 0a 10 02: 变量数据排列格式 推荐 12 0a 10 02 12 0a 10 01: 位排列 12 0a 10 02: 字节排列 12 0a 10 04: 字排列 12 0a 10 06: 双字排列
00 01: 变量数目 [1x1=1byte] 根据变量数据排列格式和变量字节数确定数目
00 00: PLC 存储区地址 根据具体存储区确定 DB Section Adr/256 Adr%256 V Section 00 01 other Sections 00 00
83: PLC 存储区 根据具体存储区确定



I Section 81
Q Section 82
M Section 83
DB Section 84
V Section 84

00 00 00: 包偏移地址 + 位偏移地址

根据具体偏移地址确定
(包偏移地址 + 位偏移地址)*8/0x10000
((包偏移地址 + 位偏移地址)*8%0x10000)/256
(包偏移地址 + 位偏移地址)*8%0x10000%256

00 04: 位标志

00 03: – 位操作
00 04: – 非位操作

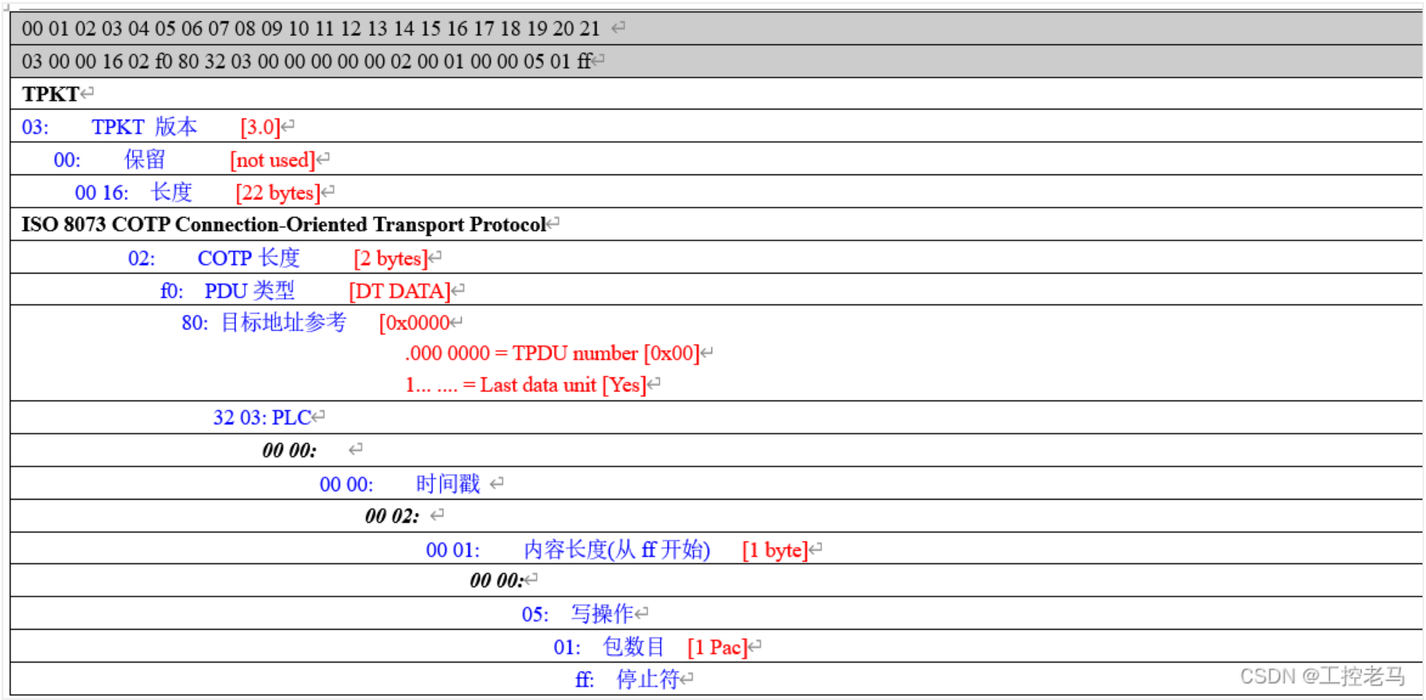
00 08: 变量长度 [8 Bit]

按位计

0a 00: 变量值

//---

[PLC -> PC]



```
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21
03 00 00 16 02 f0 80 32 03 00 00 00 00 00 02 00 01 00 00 05 01 ff
```

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 16: 长度 [22 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

02: COTP 长度 [2 bytes]

f0: PDU 类型 [DT DATA]

80: 目标地址参考 [0x0000
.000 0000 = TPDU number [0x00]
1... .... = Last data unit [Yes]

32 03: PLC

00 00:

00 00: 时间戳

00 02:

00 01: 内容长度(从 ff 开始) [1 byte]

00 00:

05: 写操作

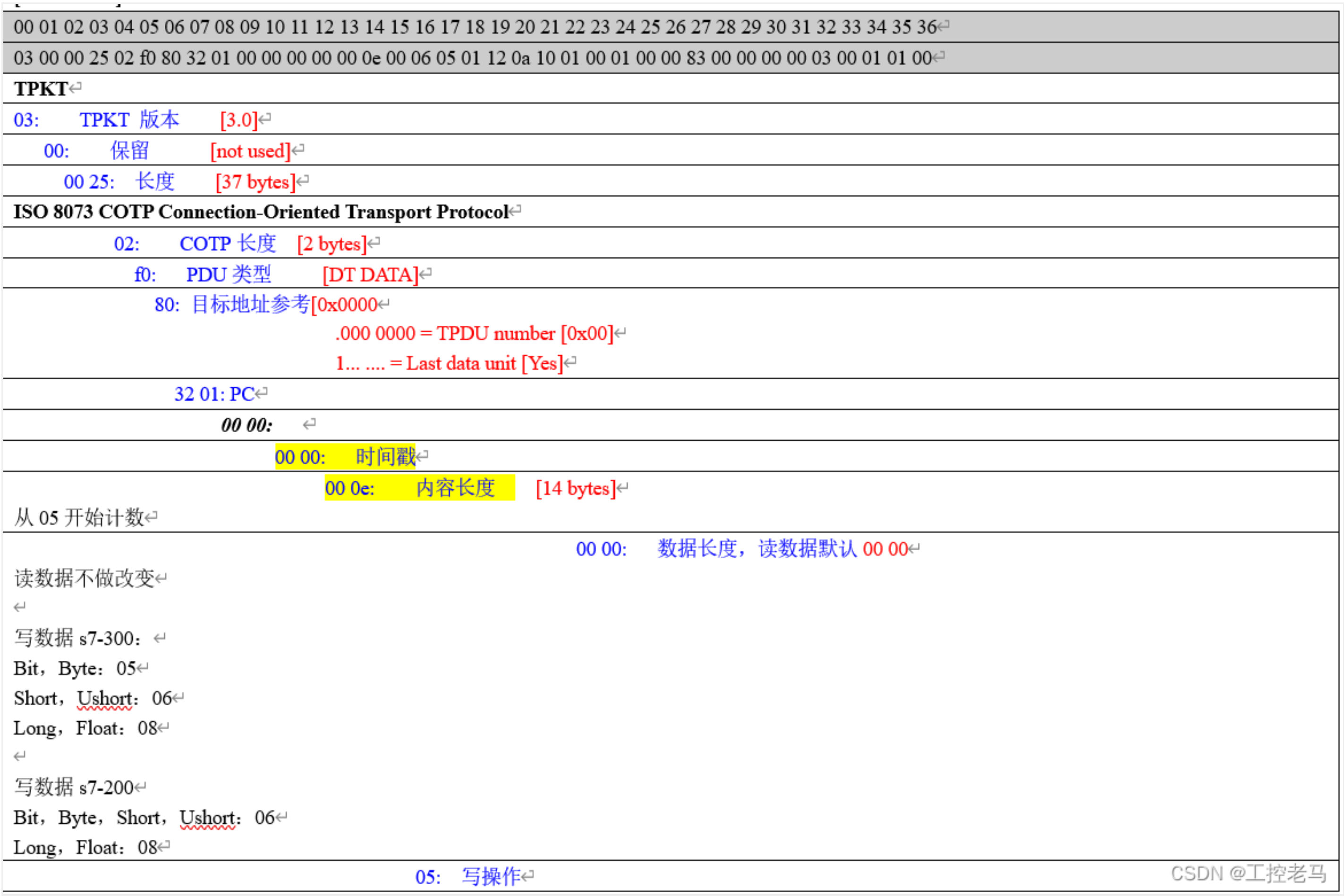
01: 包数目 [1 Pac]

ff: 停止符

4.1.2 Reference example [Q0.0]

//---

[PC -> PLC]



```
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
03 00 00 25 02 f0 80 32 01 00 00 00 00 00 0e 00 06 05 01 12 0a 10 01 00 01 00 00 83 00 00 00 00 03 00 01 01 00
```

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 25: 长度 [37 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

02: COTP 长度 [2 bytes]

f0: PDU 类型 [DT DATA]

80: 目标地址参考[0x0000
.000 0000 = TPDU number [0x00]
1... .... = Last data unit [Yes]

32 01: PC

00 00:

00 00: 时间戳

00 0e: 内容长度 [14 bytes]

从 05 开始计数

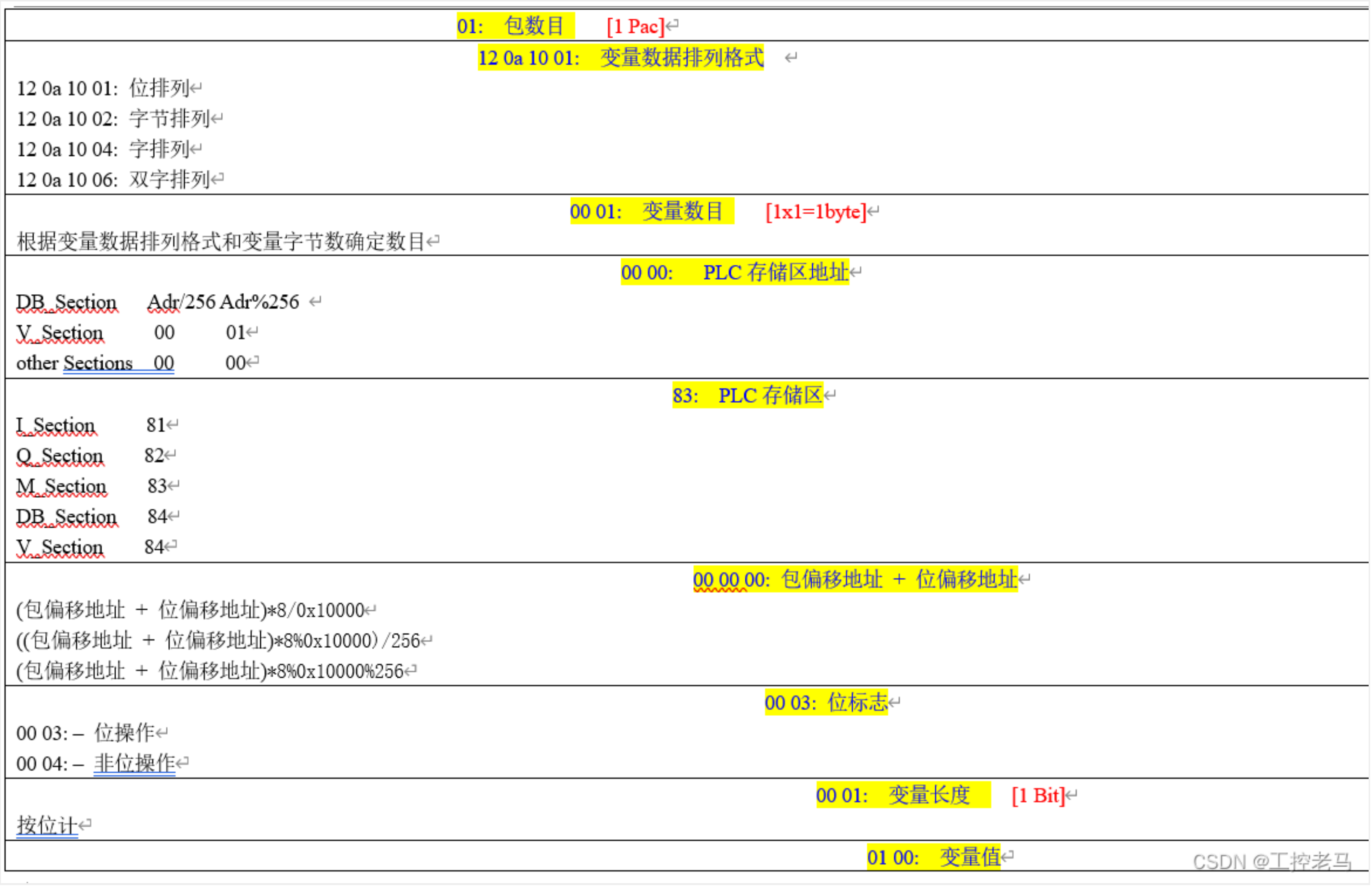
00 00: 数据长度，读数据默认 00 00

读数据不做改变

写数据 s7-300：
Bit，Byte：05
Short，Ushort：06
Long，Float：08

写数据 s7-200
Bit，Byte，Short，Ushort：06
Long，Float：08

05: 写操作



01: 包数目 [1 Pac]

12 0a 10 01: 变量数据排列格式

12 0a 10 01: 位排列
12 0a 10 02: 字节排列
12 0a 10 04: 字排列
12 0a 10 06: 双字排列

00 01: 变量数目 [1x1=1byte]

根据变量数据排列格式和变量字节数确定数目

00 00: PLC 存储区地址

DB Section	Adr/256	Adr%256
V Section	00	01
other Sections	00	00

83: PLC 存储区

I Section	81
Q Section	82
M Section	83
DB Section	84
V Section	84

00 00 00: 包偏移地址 + 位偏移地址

(包偏移地址 + 位偏移地址)*8/0x10000
((包偏移地址 + 位偏移地址)*8%0x10000)/256
(包偏移地址 + 位偏移地址)*8%0x10000%256

00 03: 位标志

00 03: – 位操作
00 04: – 非位操作

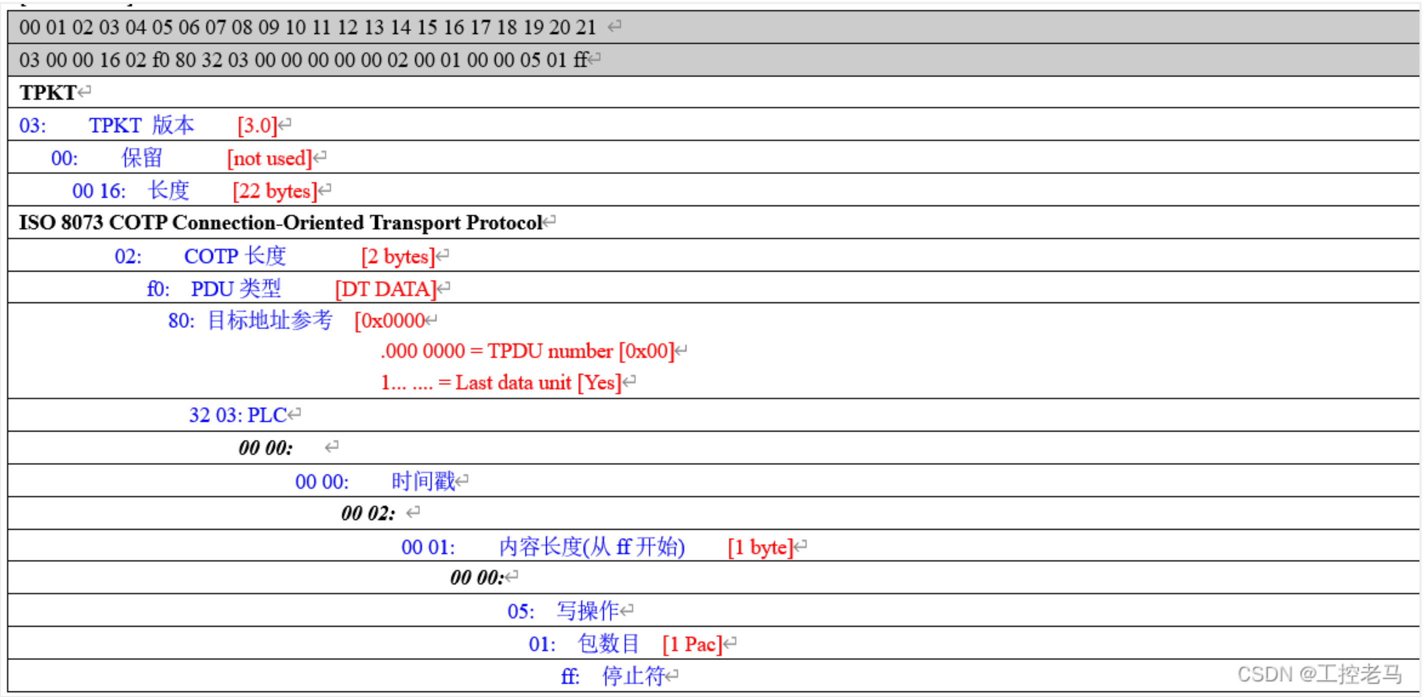
00 01: 变量长度 [1 Bit]

按位计

01 00: 变量值

//---

[PLC -> PC]



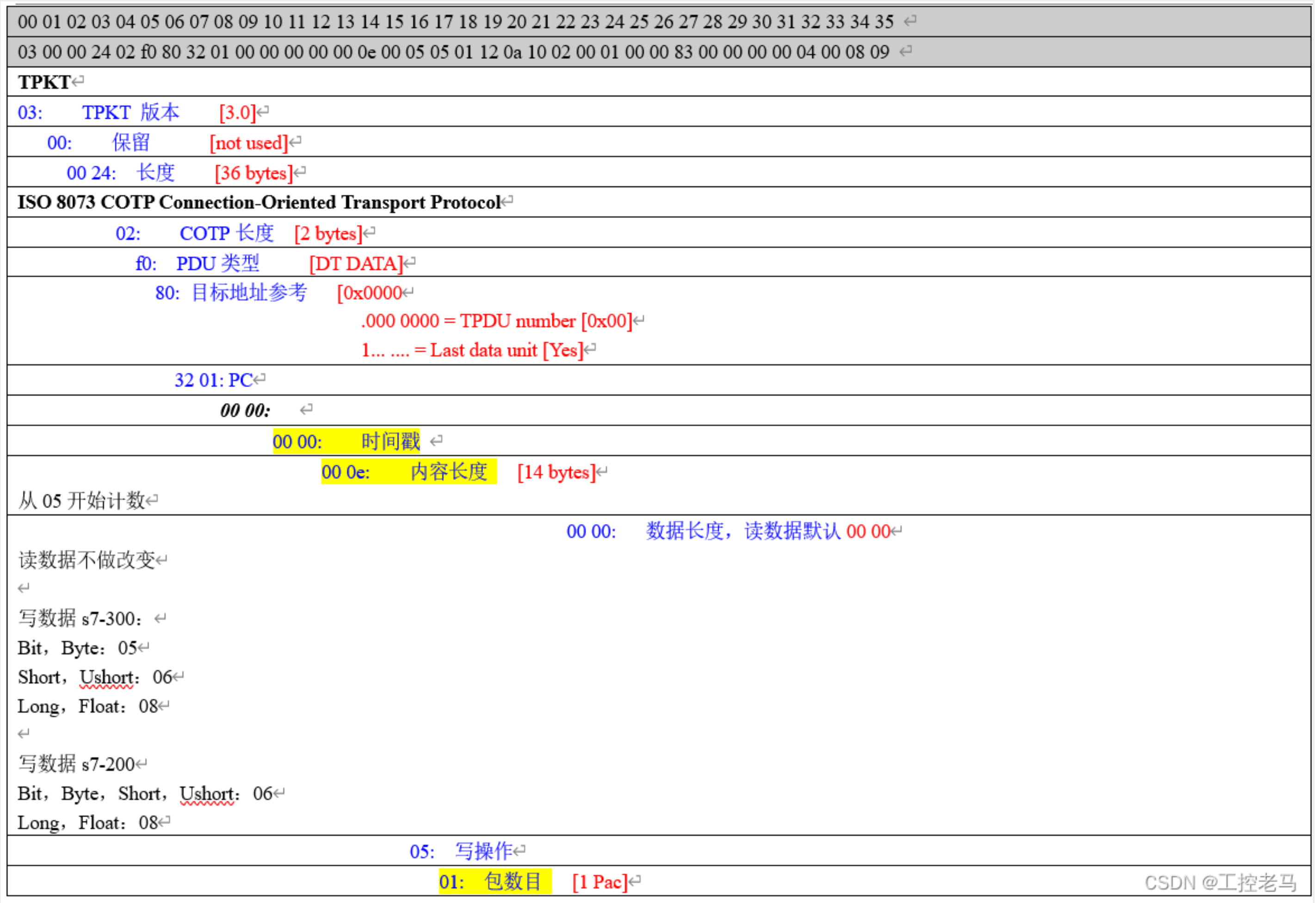
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 ↵
03 00 00 16 02 f0 80 32 03 00 00 00 00 00 02 00 01 00 00 05 01 ff↵
TPKT↵
03: TPKT 版本 [3.0]↵
00: 保留 [not used]↵
00 16: 长度 [22 bytes]↵
ISO 8073 COTP Connection-Oriented Transport Protocol↵
02: COTP 长度 [2 bytes]↵
f0: PDU 类型 [DT DATA]↵
80: 目标地址参考 [0x0000↵ .000 0000 = TPDU number [0x00]↵ 1... .... = Last data unit [Yes]↵
32 03: PLC↵
00 00: ↵
00 00: 时间戳↵
00 02: ↵
00 01: 内容长度(从 ff 开始) [1 byte]↵
00 00:↵
05: 写操作↵
01: 包数目 [1 Pac]↵
ff: 停止符↵

4.2 S7-300

4.2.1 Typical Example [MB0]

//---

[PC -> PLC]



```
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35
03 00 00 24 02 f0 80 32 01 00 00 00 00 00 0e 00 05 05 01 12 0a 10 02 00 01 00 00 83 00 00 00 00 04 00 08 09
```

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 24: 长度 [36 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

02: COTP 长度 [2 bytes]

f0: PDU 类型 [DT DATA]

80: 目标地址参考 [0x0000
.000 0000 = TPDU number [0x00]
1... .... = Last data unit [Yes]

32 01: PC

00 00:

00 00: 时间戳

00 0e: 内容长度 [14 bytes]

从 05 开始计数

00 00: 数据长度，读数据默认 00 00

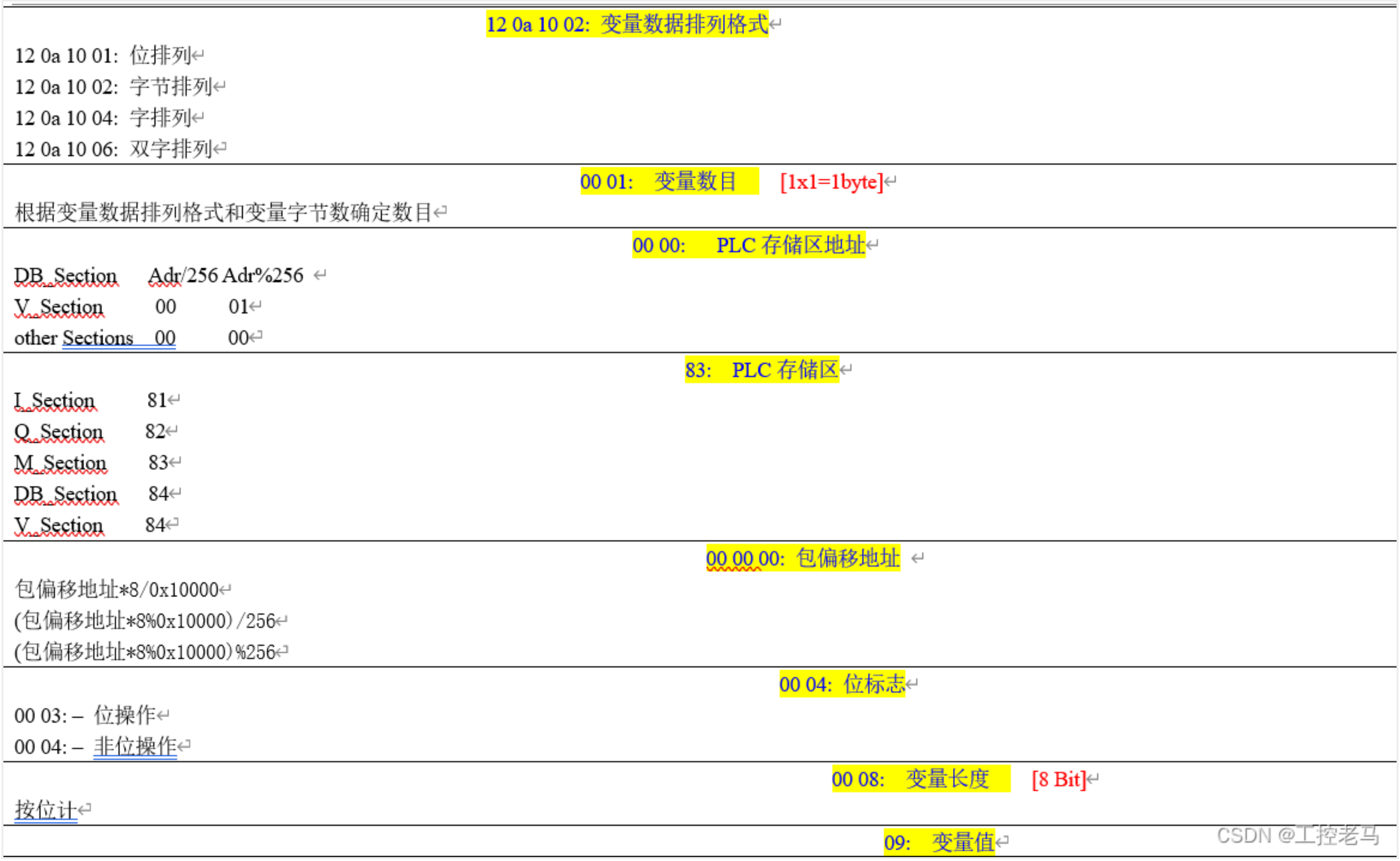
读数据不做改变

写数据 s7-300：
Bit，Byte：05
Short，Ushort：06
Long，Float：08

写数据 s7-200
Bit，Byte，Short，Ushort：06
Long，Float：08

05: 写操作

01: 包数目 [1 Pac]



12 0a 10 02: 变量数据排列格式

12 0a 10 01: 位排列
12 0a 10 02: 字节排列
12 0a 10 04: 字排列
12 0a 10 06: 双字排列

00 01: 变量数目 [1x1=1byte]

根据变量数据排列格式和变量字节数确定数目

00 00: PLC 存储区地址

DB Section	Adr/256	Adr%256
V Section	00	01
other Sections	00	00

83: PLC 存储区

I Section	81
Q Section	82
M Section	83
DB Section	84
V Section	84

00 00 00: 包偏移地址

包偏移地址*8/0x10000
(包偏移地址*8%0x10000)/256
(包偏移地址*8%0x10000)%256

00 04: 位标志

00 03: – 位操作
00 04: – 非位操作

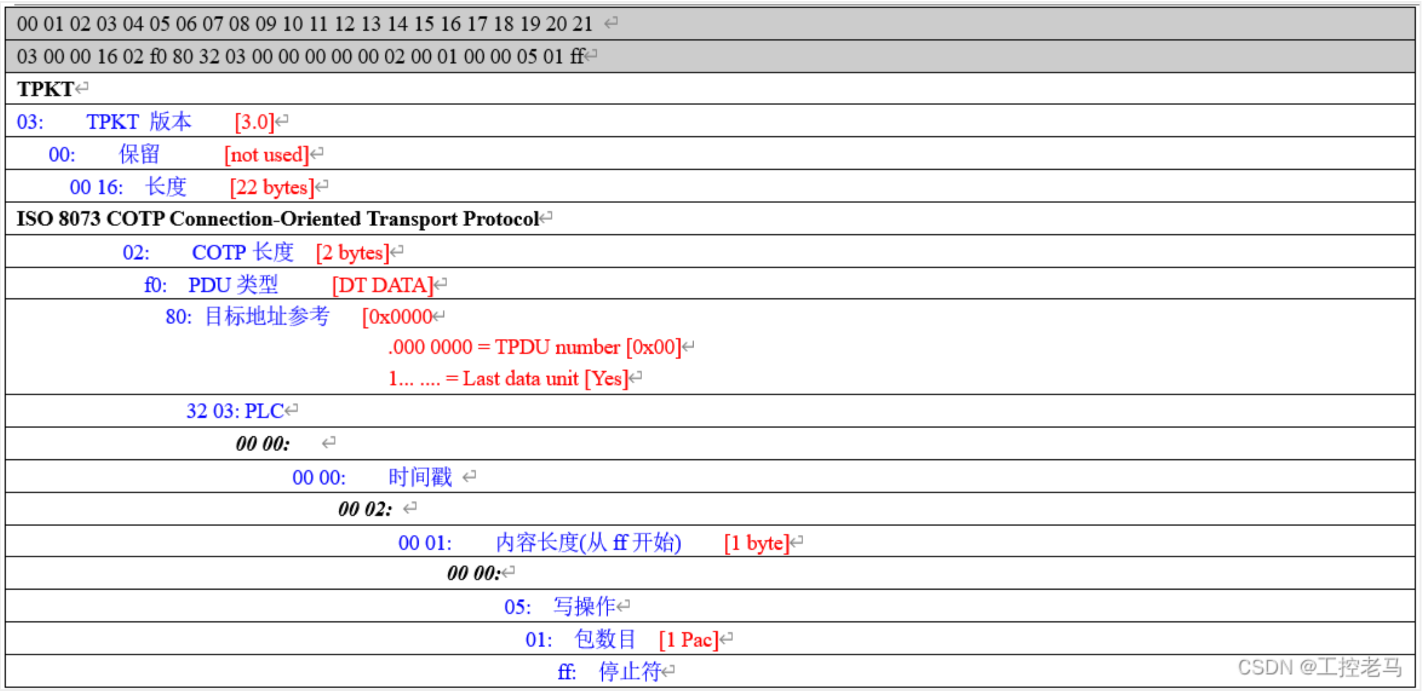
00 08: 变量长度 [8 Bit]

按位计

09: 变量值

//---

[PLC -> PC]

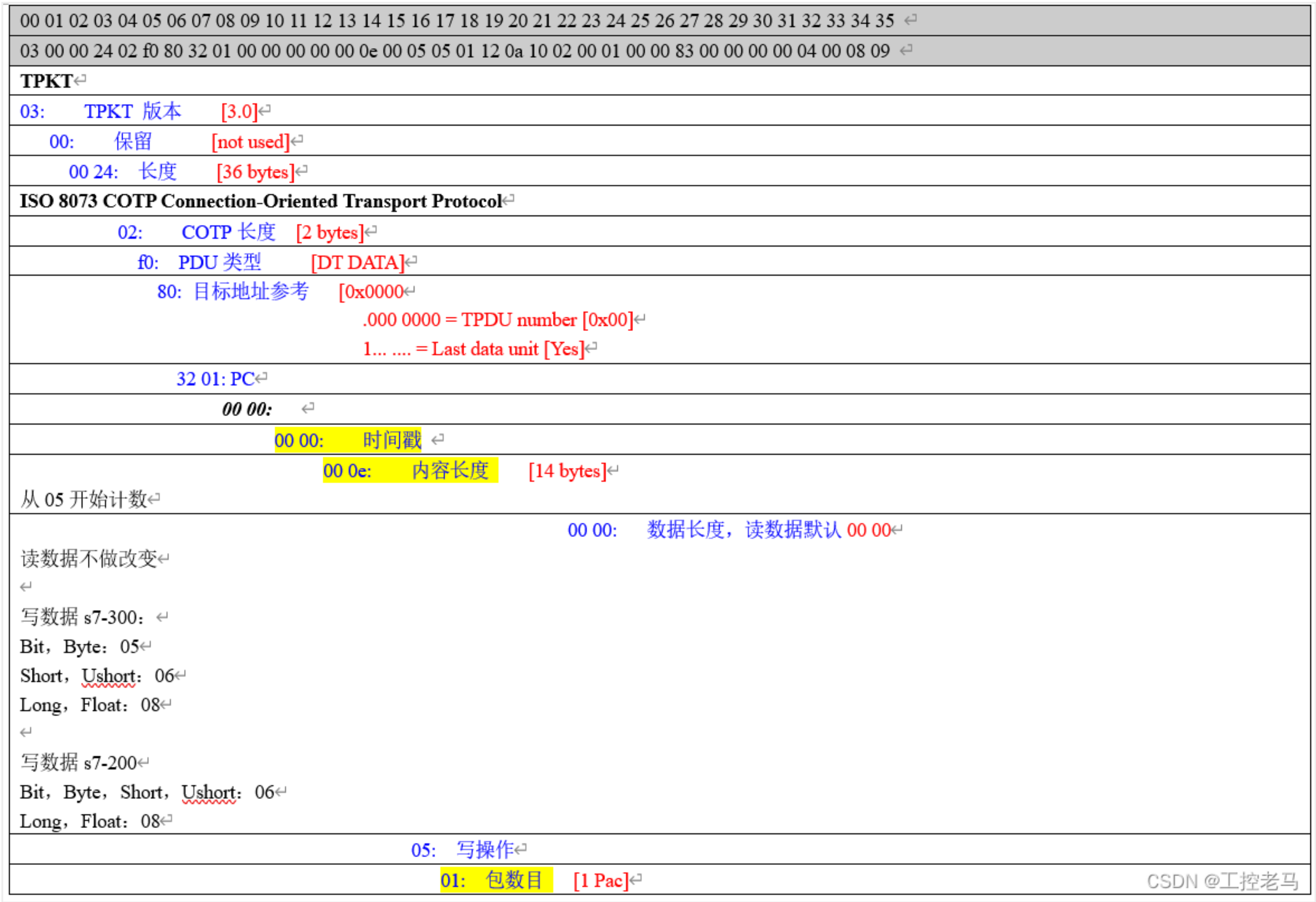


00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 ↵
03 00 00 16 02 f0 80 32 03 00 00 00 00 00 02 00 01 00 00 05 01 ff↵
TPKT↵
03: TPKT 版本 [3.0]↵
00: 保留 [not used]↵
00 16: 长度 [22 bytes]↵
ISO 8073 COTP Connection-Oriented Transport Protocol↵
02: COTP 长度 [2 bytes]↵
f0: PDU 类型 [DT DATA]↵
80: 目标地址参考 [0x0000↵ .000 0000 = TPDU number [0x00]↵ 1... .... = Last data unit [Yes]↵
32 03: PLC↵
00 00: ↵
00 00: 时间戳 ↵
00 02: ↵
00 01: 内容长度(从 ff 开始) [1 byte]↵
00 00:↵
05: 写操作↵
01: 包数目 [1 Pac]↵
ff: 停止符↵

4.2.2 Typical Example [M0.3]

//---

[PC -> PLC]



```
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35
03 00 00 24 02 f0 80 32 01 00 00 00 00 00 0e 00 05 05 01 12 0a 10 02 00 01 00 00 83 00 00 00 00 04 00 08 09
```

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 24: 长度 [36 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

02: COTP 长度 [2 bytes]

f0: PDU 类型 [DT DATA]

80: 目标地址参考 [0x0000

.000 0000 = TPDU number [0x00]

1... .... = Last data unit [Yes]

32 01: PC

00 00:

00 00: 时间戳

00 0e: 内容长度 [14 bytes]

从 05 开始计数

00 00: 数据长度，读数据默认 00 00

读数据不做改变

写数据 s7-300：

Bit，Byte：05

Short，Ushort：06

Long，Float：08

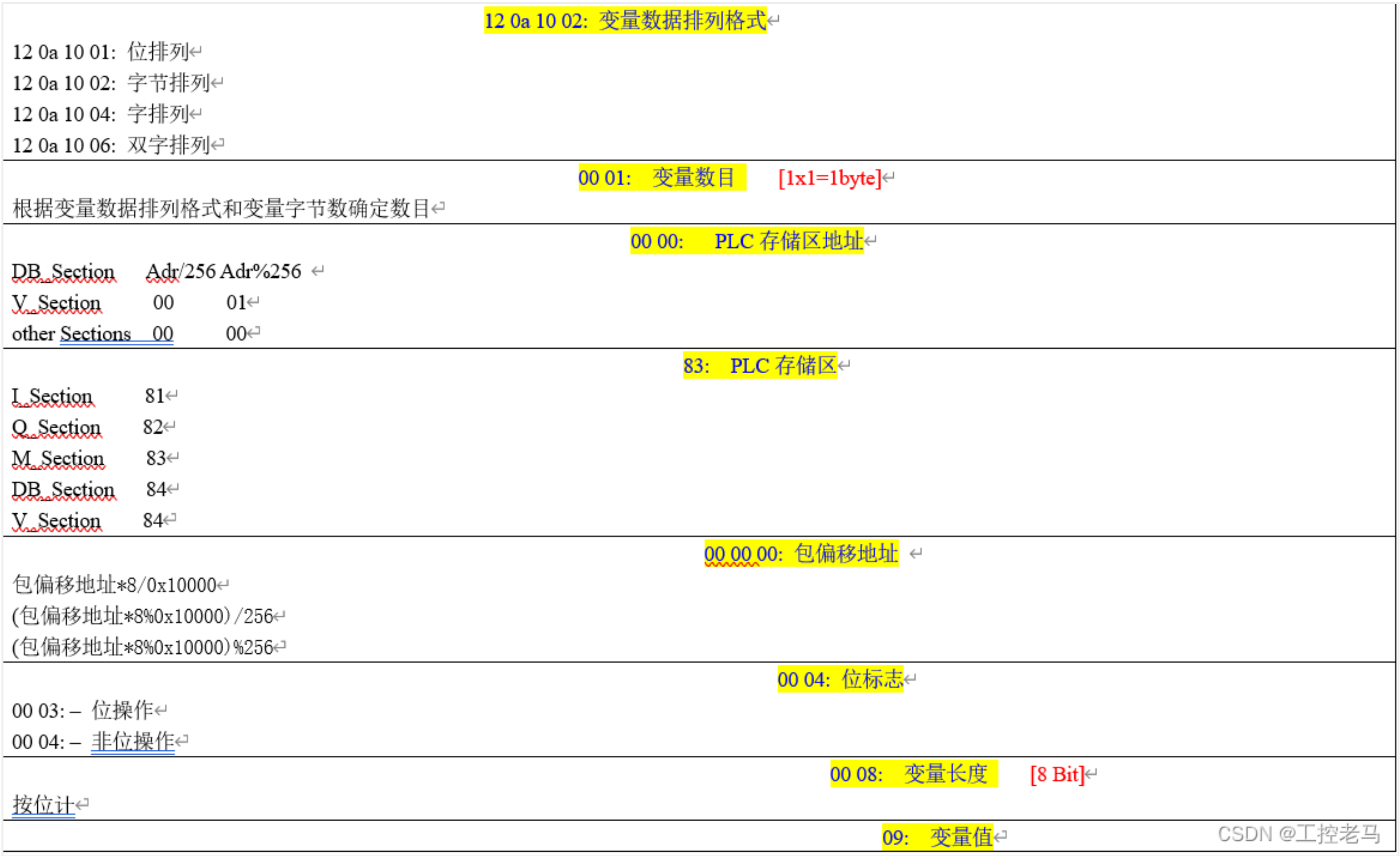
写数据 s7-200

Bit，Byte，Short，Ushort：06

Long，Float：08

05: 写操作

01: 包数目 [1 Pac]



12 0a 10 02: 变量数据排列格式

12 0a 10 01: 位排列

12 0a 10 02: 字节排列

12 0a 10 04: 字排列

12 0a 10 06: 双字排列

00 01: 变量数目 [1x1=1byte]

根据变量数据排列格式和变量字节数确定数目

00 00: PLC 存储区地址

DB_Section	Adr/256	Adr%256
V_Section	00	01
other Sections	00	00

83: PLC 存储区

I_Section	81
Q_Section	82
M_Section	83
DB_Section	84
V_Section	84

00 00 00: 包偏移地址

包偏移地址*8/0x10000

(包偏移地址*8%0x10000)/256

(包偏移地址*8%0x10000)%256

00 04: 位标志

00 03: – 位操作

00 04: – 非位操作

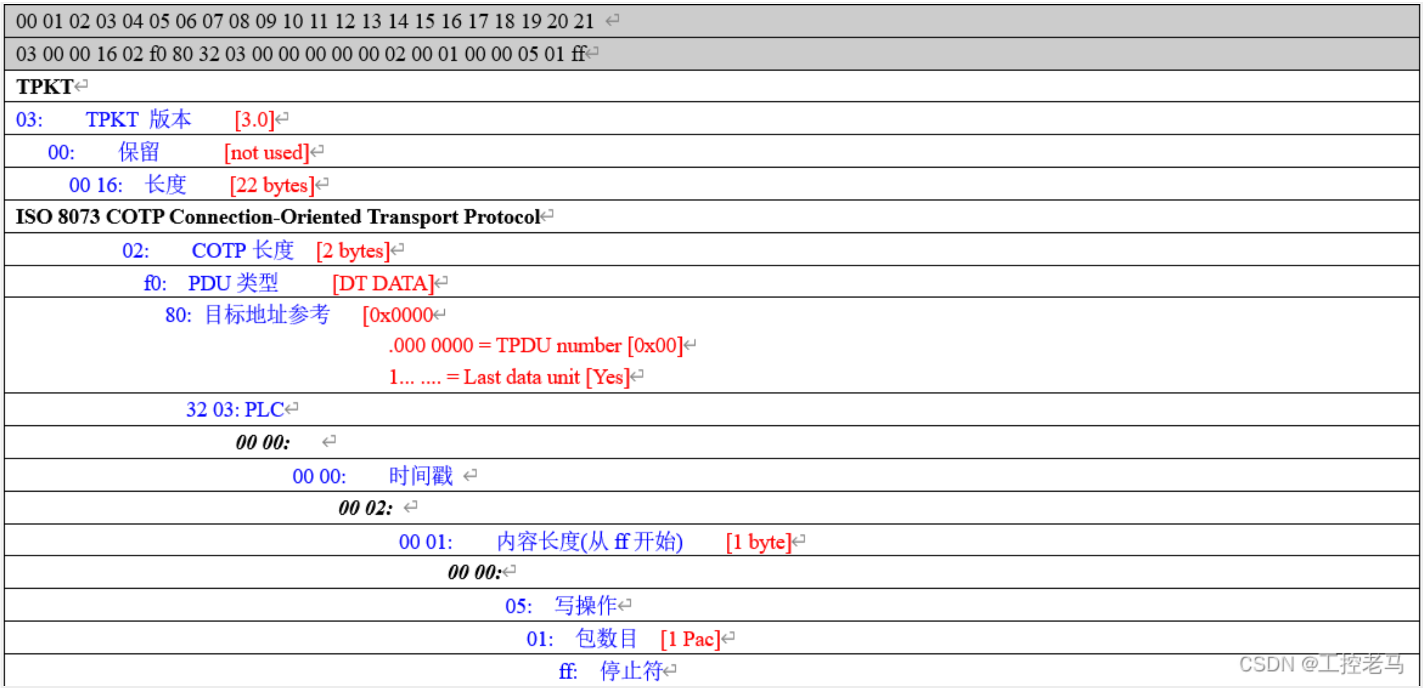
00 08: 变量长度 [8 Bit]

按位计

09: 变量值

//---

[PLC -> PC]

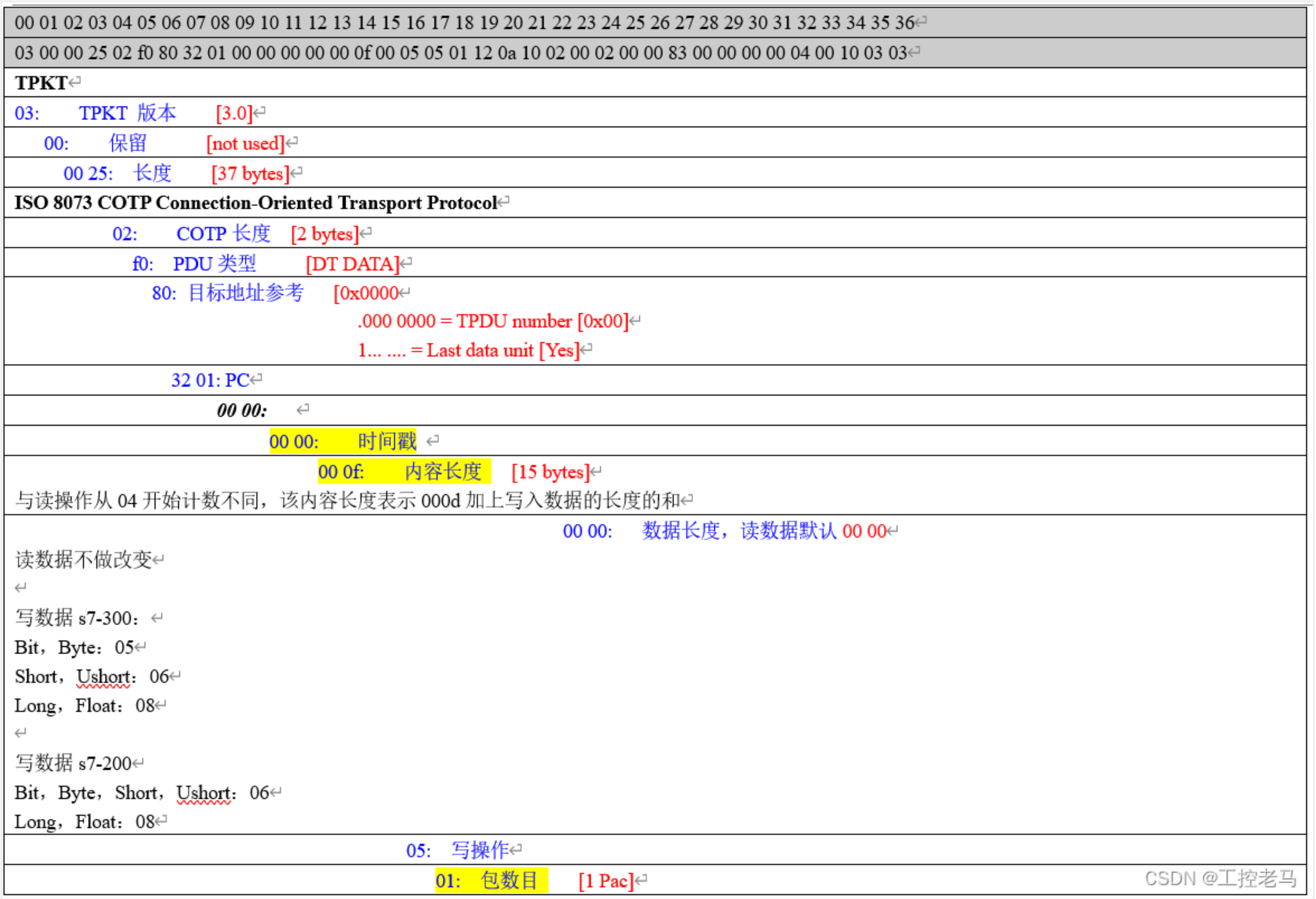


00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21
03 00 00 16 02 f0 80 32 03 00 00 00 00 00 02 00 01 00 00 05 01 ff
TPKT
03: TPKT 版本 [3.0]
00: 保留 [not used]
00 16: 长度 [22 bytes]
ISO 8073 COTP Connection-Oriented Transport Protocol
02: COTP 长度 [2 bytes]
f0: PDU 类型 [DT DATA]
80: 目标地址参考 [0x0000 .000 0000 = TPDU number [0x00] 1... .... = Last data unit [Yes]
32 03: PLC
00 00:
00 00: 时间戳
00 02:
00 01: 内容长度(从 ff 开始) [1 byte]
00 00:
05: 写操作
01: 包数目 [1 Pac]
ff: 停止符

4.2.3 Typical Example [MW0]

//---

[PC -> PLC]



00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36

03 00 00 25 02 f0 80 32 01 00 00 00 00 00 0f 00 05 05 01 12 0a 10 02 00 02 00 00 83 00 00 00 00 04 00 10 03 03

TPKT

03: TPKT 版本 [3.0]

00: 保留 [not used]

00 25: 长度 [37 bytes]

ISO 8073 COTP Connection-Oriented Transport Protocol

02: COTP 长度 [2 bytes]

f0: PDU 类型 [DT DATA]

80: 目标地址参考 [0x0000

.000 0000 = TPDU number [0x00]

1... .... = Last data unit [Yes]

32 01: PC

00 00:

00 00: 时间戳

00 0f: 内容长度 [15 bytes]

与读操作从 04 开始计数不同，该内容长度表示 000d 加上写入数据的长度的和

00 00: 数据长度，读数据默认 00 00

读数据不做改变

写数据 s7-300：

Bit，Byte：05

Short，Ushort：06

Long，Float：08

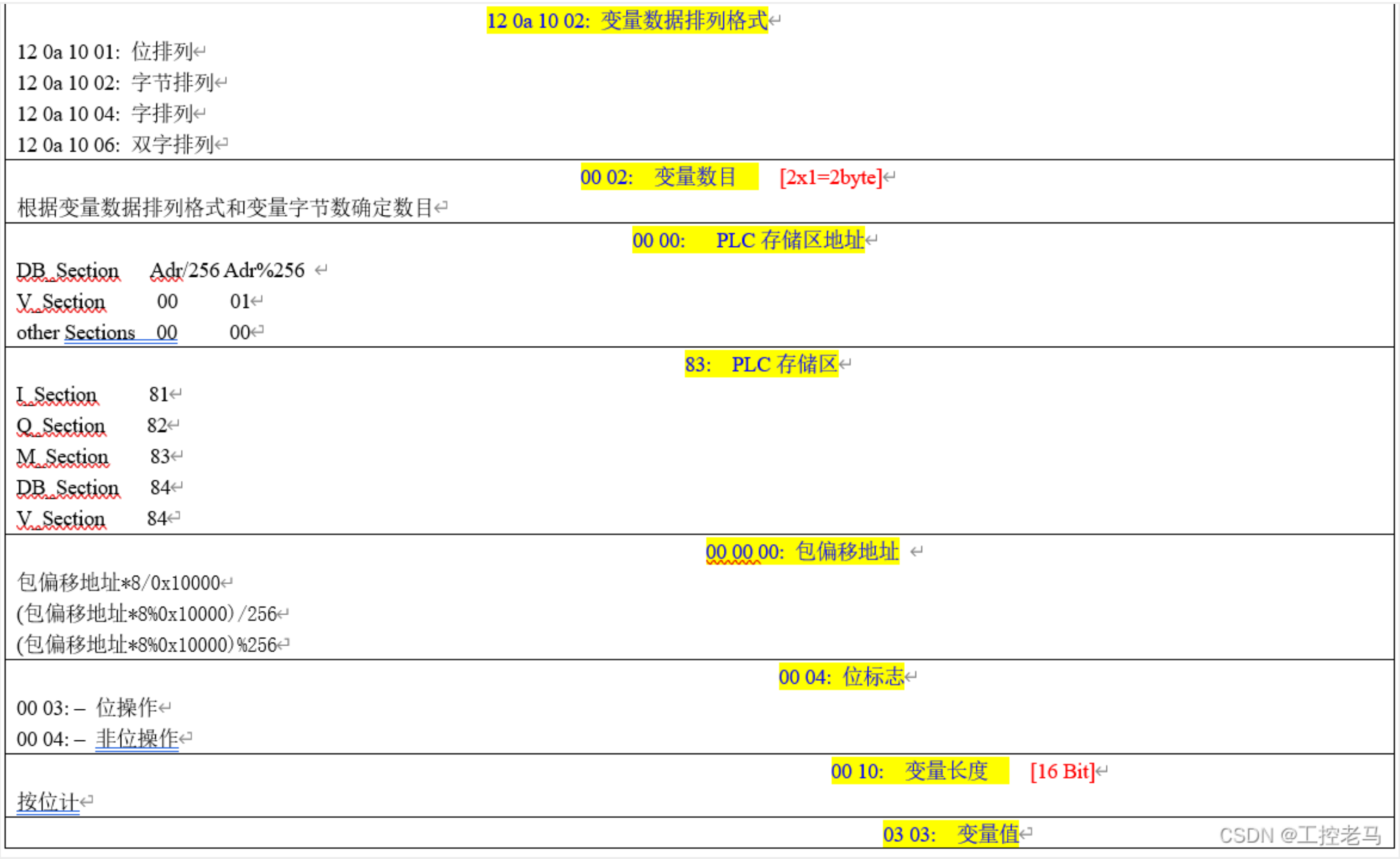
写数据 s7-200

Bit，Byte，Short，Ushort：06

Long，Float：08

05: 写操作

01: 包数目 [1 Pac]



12 0a 10 02: 变量数据排列格式

12 0a 10 01: 位排列

12 0a 10 02: 字节排列

12 0a 10 04: 字排列

12 0a 10 06: 双字排列

00 02: 变量数目 [2x1=2byte]

根据变量数据排列格式和变量字节数确定数目

00 00: PLC 存储区地址

DB Section	Adr/256	Adr%256
V Section	00	01
other Sections	00	00

83: PLC 存储区

I Section	81
Q Section	82
M Section	83
DB Section	84
V Section	84

00 00 00: 包偏移地址

包偏移地址*8/0x10000

(包偏移地址*8%0x10000)/256

(包偏移地址*8%0x10000)%256

00 04: 位标志

00 03: – 位操作

00 04: – 非位操作

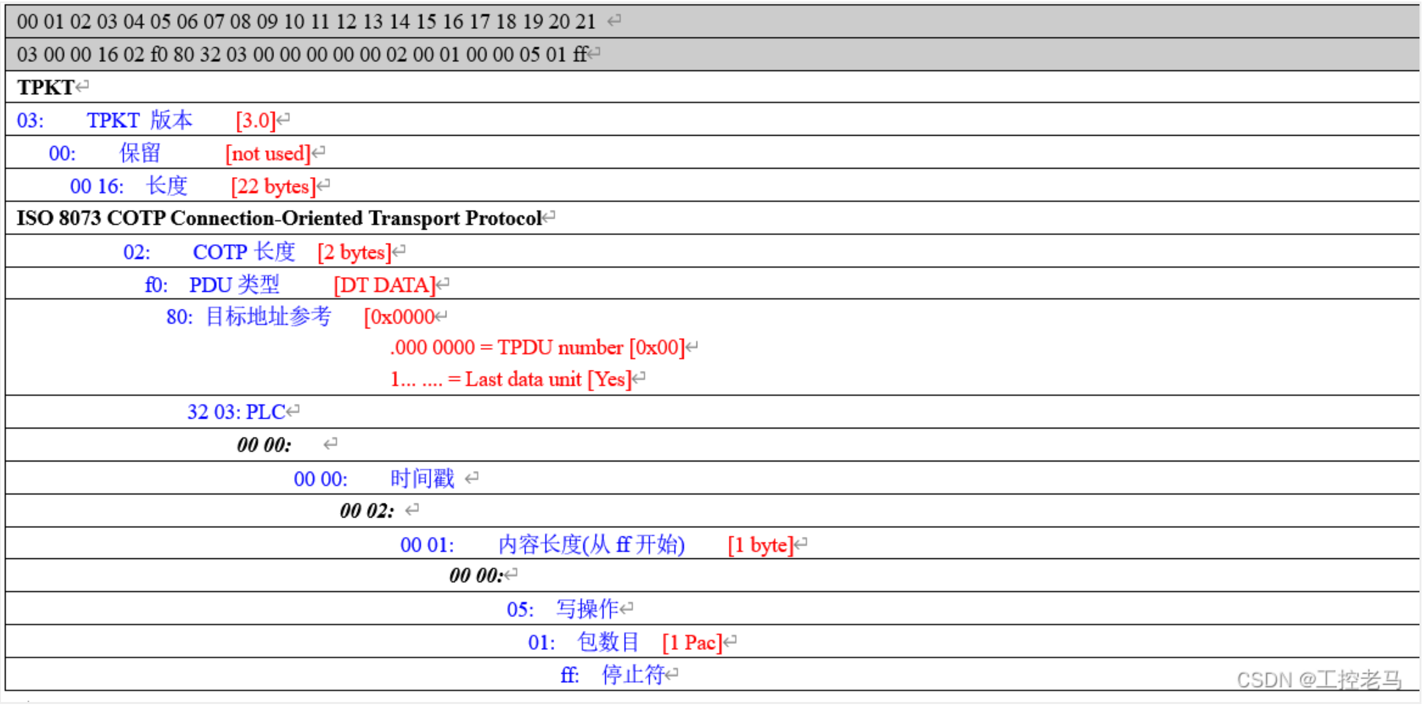
00 10: 变量长度 [16 Bit]

按位计

03 03: 变量值

//---

[PLC -> PC]



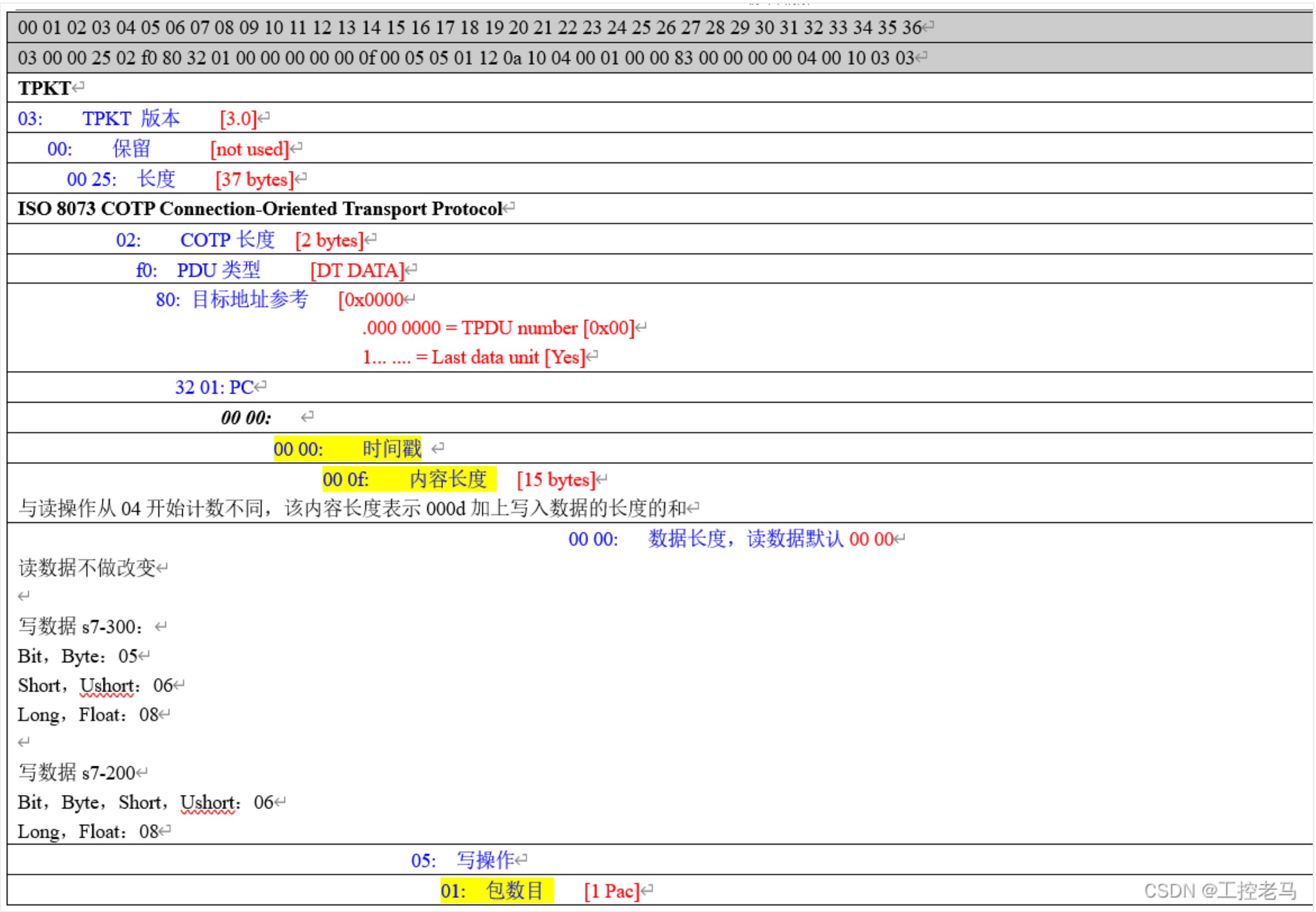
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21
03 00 00 16 02 f0 80 32 03 00 00 00 00 00 02 00 01 00 00 05 01 ff
TPKT
03: TPKT 版本 [3.0]
00: 保留 [not used]
00 16: 长度 [22 bytes]
ISO 8073 COTP Connection-Oriented Transport Protocol
02: COTP 长度 [2 bytes]
f0: PDU 类型 [DT DATA]
80: 目标地址参考 [0x0000 .000 0000 = TPDU number [0x00] 1... .... = Last data unit [Yes]
32 03: PLC
00 00:
00 00: 时间戳
00 02:
00 01: 内容长度(从 ff 开始) [1 byte]
00 00:
05: 写操作
01: 包数目 [1 Pac]
ff: 停止符

CSDN @工控老马

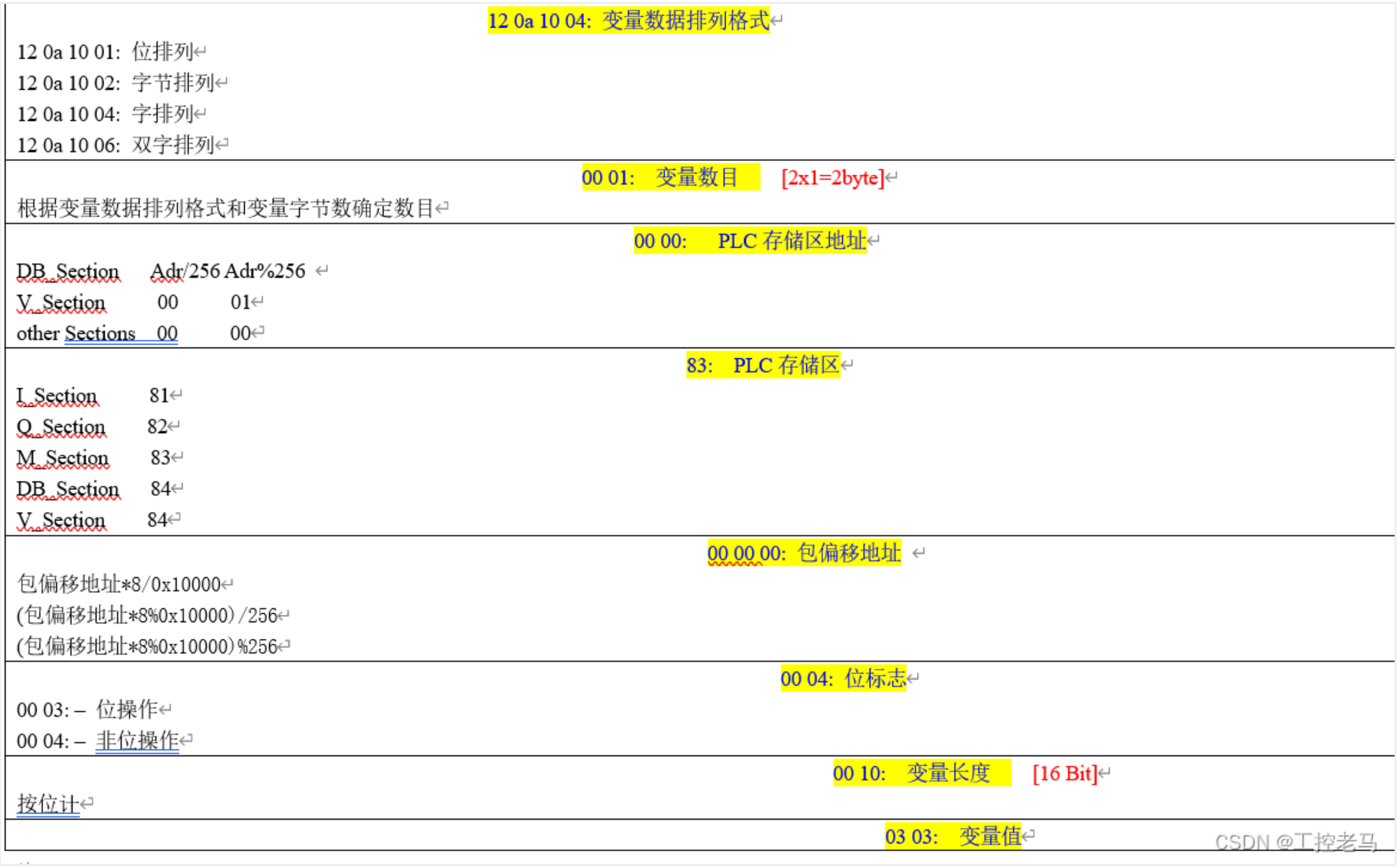
4.2.4 Typical Example [MW0] Another protocol frame, the difference lies in the variable data arrangement format and the number of variables

//---

[PC -> PLC]



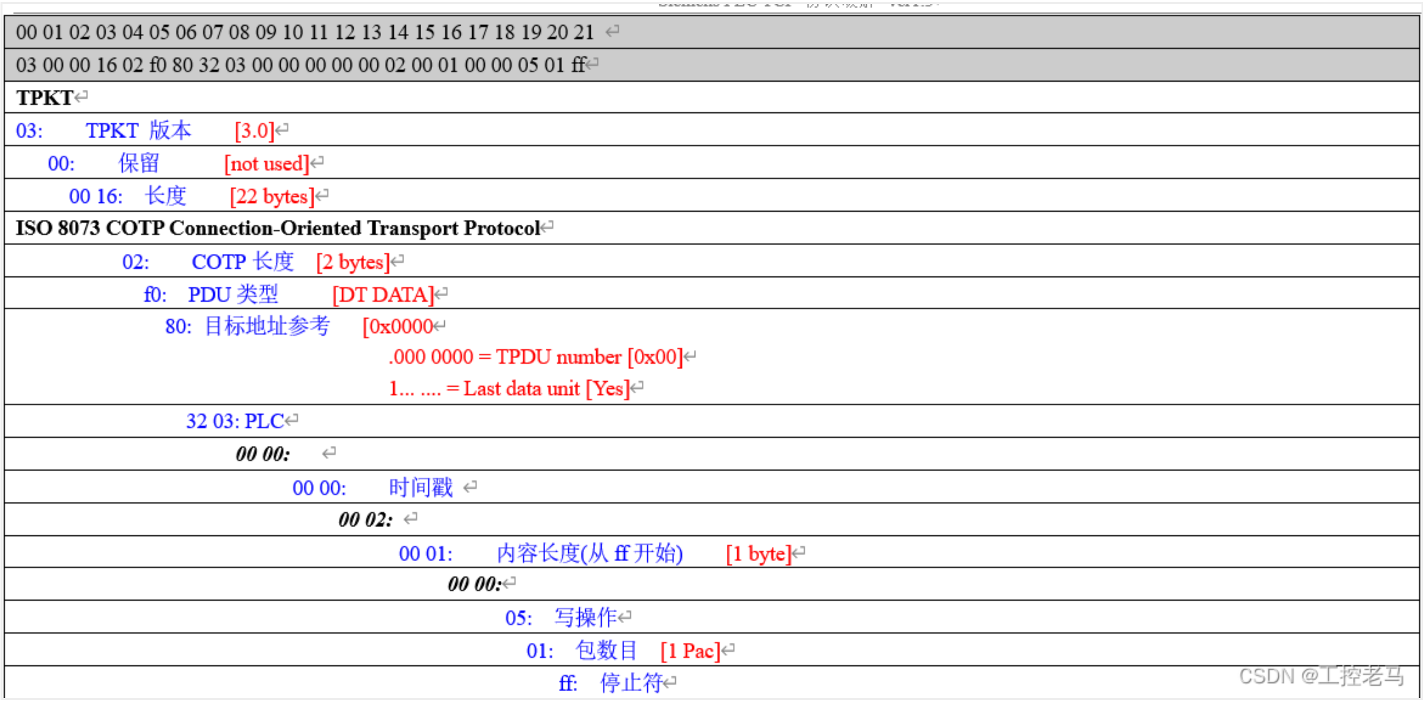
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
03 00 00 25 02 f0 80 32 01 00 00 00 00 00 0f 00 05 05 01 12 0a 10 04 00 01 00 00 83 00 00 00 00 04 00 10 03 03
TPKT
03: TPKT 版本 [3.0]
00: 保留 [not used]
00 25: 长度 [37 bytes]
ISO 8073 COTP Connection-Oriented Transport Protocol
02: COTP 长度 [2 bytes]
f0: PDU 类型 [DT DATA]
80: 目标地址参考 [0x0000 .000 0000 = TPDU number [0x00] 1... .... = Last data unit [Yes]
32 01: PC
00 00:
00 00: 时间戳
00 0f: 内容长度 [15 bytes] 与读操作从 04 开始计数不同，该内容长度表示 000d 加上写入数据的长度的和
00 00: 数据长度，读数据默认 00 00 读数据不做改变 写数据 s7-300： Bit，Byte：05 Short，Ushort：06 Long，Float：08 写数据 s7-200 Bit，Byte，Short，Ushort：06 Long，Float：08
05: 写操作
01: 包数目 [1 Pac]



12 0a 10 04: 变量数据排列格式 12 0a 10 01: 位排列 12 0a 10 02: 字节排列 12 0a 10 04: 字排列 12 0a 10 06: 双字排列
00 01: 变量数目 [2x1=2byte] 根据变量数据排列格式和变量字节数确定数目
00 00: PLC 存储区地址 DB Section Adr/256 Adr%256 V Section 00 01 other Sections 00 00
83: PLC 存储区 I Section 81 Q Section 82 M Section 83 DB Section 84 V Section 84
00 00 00: 包偏移地址 包偏移地址*8/0x10000 (包偏移地址*8%0x10000)/256 (包偏移地址*8%0x10000)%256
00 04: 位标志 00 03: – 位操作 00 04: – 非位操作
00 10: 变量长度 [16 Bit] 按位计
03 03: 变量值

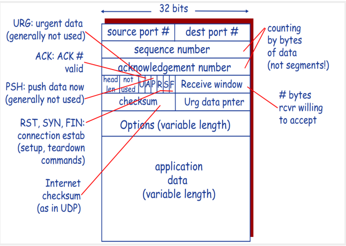
//---

[PLC -> PC]



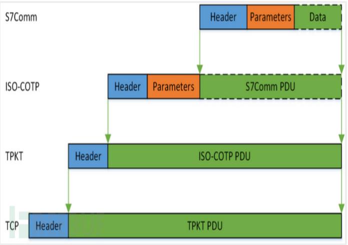
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21
03 00 00 16 02 f0 80 32 03 00 00 00 00 00 02 00 01 00 00 05 01 ff
TPKT
03: TPKT 版本 [3.0]
00: 保留 [not used]
00 16: 长度 [22 bytes]
ISO 8073 COTP Connection-Oriented Transport Protocol
02: COTP 长度 [2 bytes]
f0: PDU 类型 [DT DATA]
80: 目标地址参考 [0x0000 .000 0000 = TPDU number [0x00] 1... .... = Last data unit [Yes]
32 03: PLC
00 00:
00 00: 时间戳
00 02:
00 01: 内容长度(从 ff 开始) [1 byte]
00 00:
05: 写操作
01: 包数目 [1 Pac]
ff: 停止符

Intelligent Recommendation


32 bits
URG: urgent data (generally not used)
ACK: ACK # valid
PSH: push data now (generally not used)
RST, SYN, FIN: connection estab (setup, teardown commands)
Internet checksum (as in UDP)
source port #
dest port #
sequence number
acknowledgement number
head len
not used
U A P R S F
Receive window
checksum
Urg data pnter
Options (variable length)
application data (variable length)
counting by bytes of data (not segments!)
# bytes rcvr willing to accept

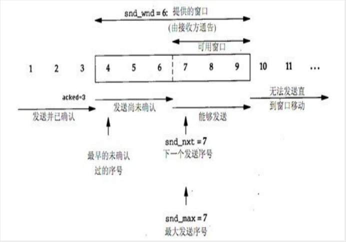
TCP protocol detailed explanation

Articles directory Foreword 1. Overview of the TCP protocol 1. Protocol features 2.TCP message segment Second, TCP reliable data transmission 1 Overview 2. Re -timeout time selection 3. Quick re -tran...



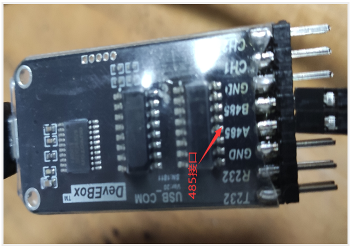
Siemens PLC protocol-S7COMM-extended

Siemens PLC protocol-S7COMM-extended Article Directory Siemens PLC protocol-S7COMM-extended S7 Communication structure TPKT protocol structure For example COTP protocol Introduction COTP connection pa...



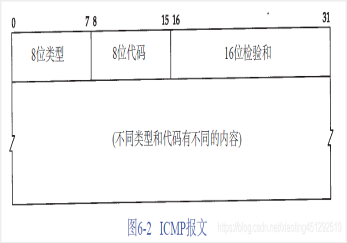
LwIP protocol stack-TCP control block (tcp_pcb) detailed explanation

The source code of the tcp_pcb structure of the TCP part of the LWIP protocol is as follows: struct tcp_pcb { IP_PCB;//This is a macro that describes the IP related information of the connection, incl...



Use QT to give Siemens PLC to Siemens PLC through the Modbusrtu protocol

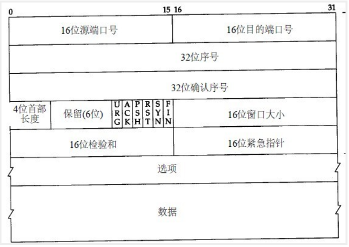
demand Use the host computer software to control the q0.0 indicator of the lower computer through the ModbusRTU protocol material computer1 USB turn 485 device1 PLC(s7-200)*1 Next machine PLC program:...



#TCP/IP# Detailed Explanation of TCP IP Volume 1: Protocol-Chapter 6 ICMP: Internet Control Message Protocol

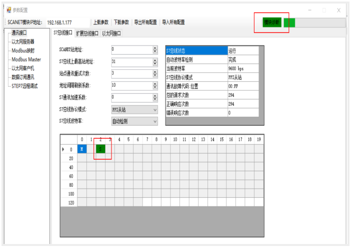
6.1 Introduction I C M P is often considered as an integral part of the I P layer. It transmits error messages and other information that needs attention. I C M P messages are usually used by IP layer...

More Recommendation



## "TCP/IP Detailed Explanation Volume 1: Protocol" Reading Notes Chapter 17 TCP: Transmission Control Protocol

Source http://blog.csdn.net/jiange_zh code> Chapter 17 TCP: Transmission Control Protocol 1. TCP services TCP provides a connection-oriented, reliable byte stream service. Connection-oriented means...

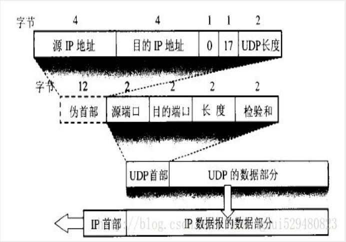


How to quickly implement Siemens S7-200/300 PLC to Modbus-TCP protocol and third-party data docking

How to quickly implement Siemens S7-200/300 PLC to Modbus-TCP protocol and third-party data docking introduction Siemens SIMATIC automation control system is widely used in the industrial control mark...

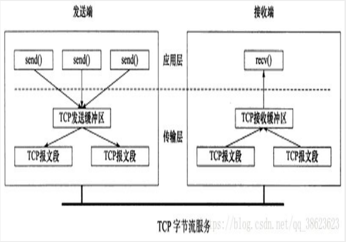
C # with Mitsubishi PLC communication detailed use (FX5U industrial control equipment)

The last article talks about any communication with Mitsubishi PLC device, this detailed introduction to the use of the project. Reading writes including 32-bit data registers. 1. Data registers (such...



Detailed explanation of TCP / UDP protocol

Both TCP and UDP protocols are transport layer protocols, a specification used to transmit data. TCP data packet format The sequence number is seq, the acknowledgment number corresponds to ack, and FI...



Detailed explanation of TCP/UDP protocol

Detailed TCP protocol The TCP connection is full-duplex, that is, both sides can read and write data through one connection. The TCP protocol connection is one-to-one, so applications based on broadca...

Related Posts

- [Industrial control old horse] Siemens PLC s7-300SCL programming detailed explanation
- [Industrial Control Old Horse] Modbus TCP Detailed Explanation

- [Industrial control old horse] Detailed design plan of marquee control system based on Siemens S7-200PLC
- 【Industrial control old horse】Detailed design of washing machine PLC program control system
- [Industrial Control Old Horse] ASCII communication guide for ABB AC500 series PLC and West 8100+ series instrument
- [Industrial control old horse] Mitsubishi Q series PLC debugging and Mitsubishi touch screen alarm instructions
- [Industrial Control Old Malays] Omron PLC Socket Send FINS / TCP Command Analysis
- Siemens PLC protocol-S7COMM
- Detailed explanation of TCP protocol (5)-TCP congestion control
- TCP flow control and congestion control-detailed explanation of sliding window protocol

Popular Posts

- Keywords: packet
- idea integrates tomcat and solves the console garbled problem
- [SOJ 639] trees
- Xiaobaixue front-end ----------------CSS basic grammar

- MYSQL date and time type format (detailed introduction)
- Network stream (template transfer)
- Layui jq finds the elements of the first element
- Python Algorithm Learning: Competitive Code Programming-Lanqiao Cup School Trial (Preliminary) Replay
- Freeswitch startup service script
- RISCV's cache

Recommended Posts

- Data Structures and Algorithms basis

- Front End Knowledge Sharing - SHEETJS Usage Experience
- Solve Endnote's reference to the problem without GBT7714 format
- Day04 (array)
- Mac installation git
- Image and large array types
- Convert grayscale image to pseudo-color image-pseudo-color processing
- GHOST blog build
- Android application component - Service
- Krpano tutorial - view tag Chinese description

Related Tags

Miscellaneous Notes on Industrial Control Technology

C++

Development language

embedded hardware

single chip microcomputer

manufacture

Industrial Control Technology Miscellaneous Records

tcp

ci

hardware engineering