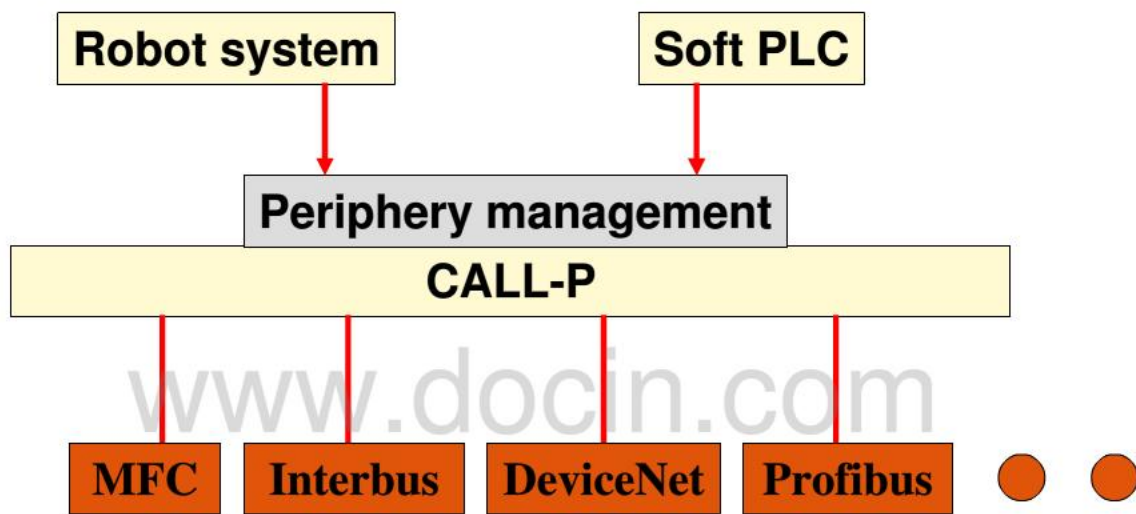
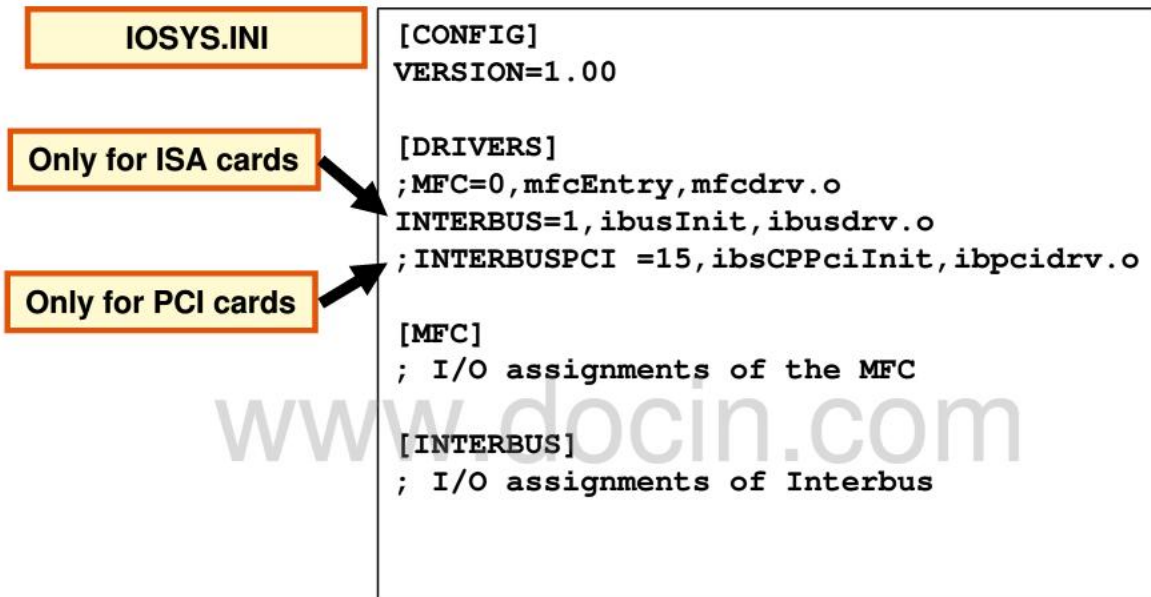
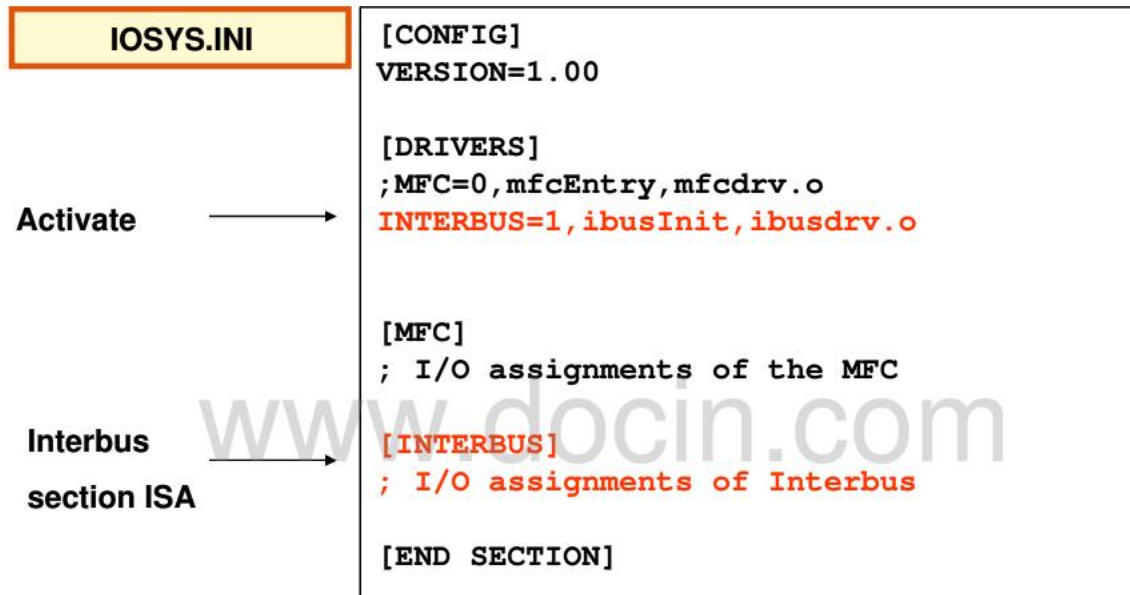
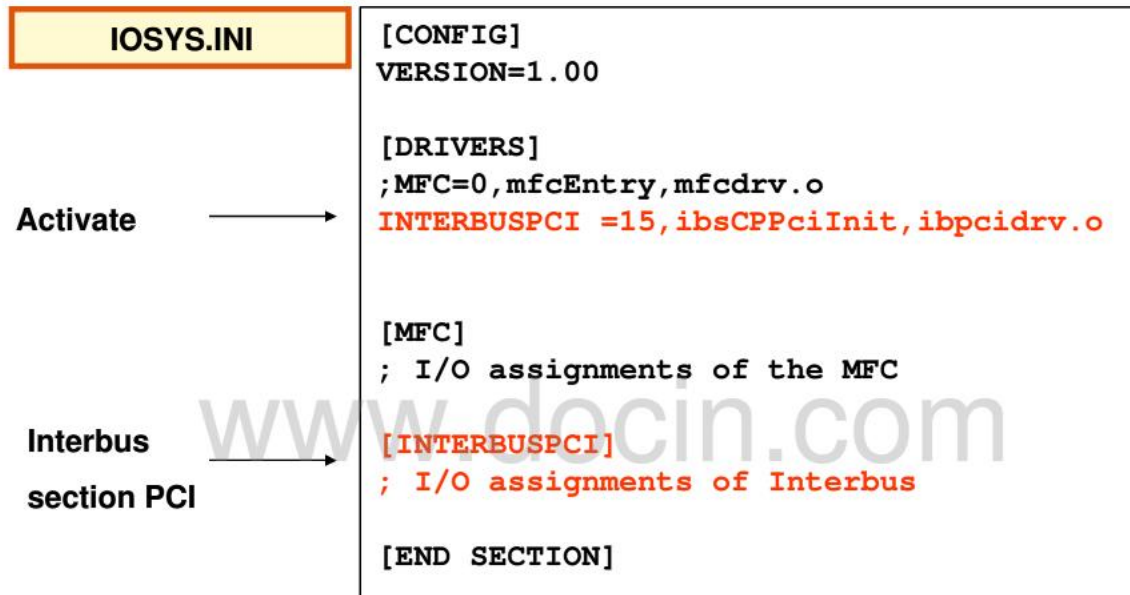
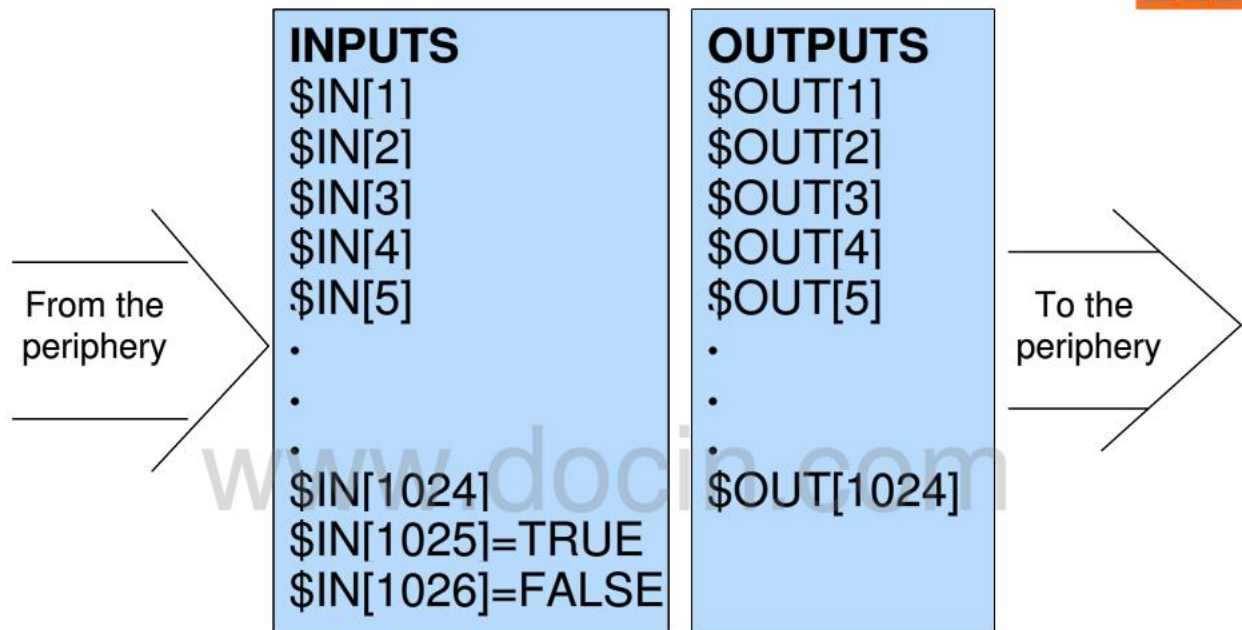




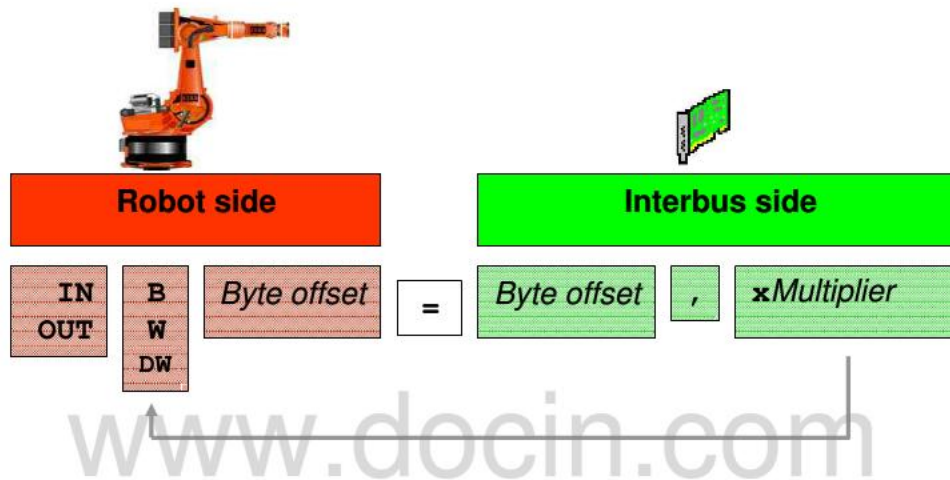








+24V=TRUE, 0V=FALSE



Example: INB2=2,x2
 OUTW6=32,x1



Text page

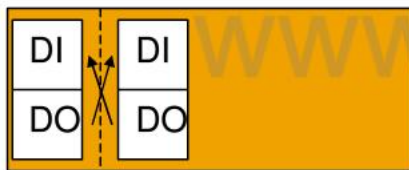
www.docin.com

If data from the **slave interface** are to be accessed, then the **offset 896** or larger must be used on the Interbus side.

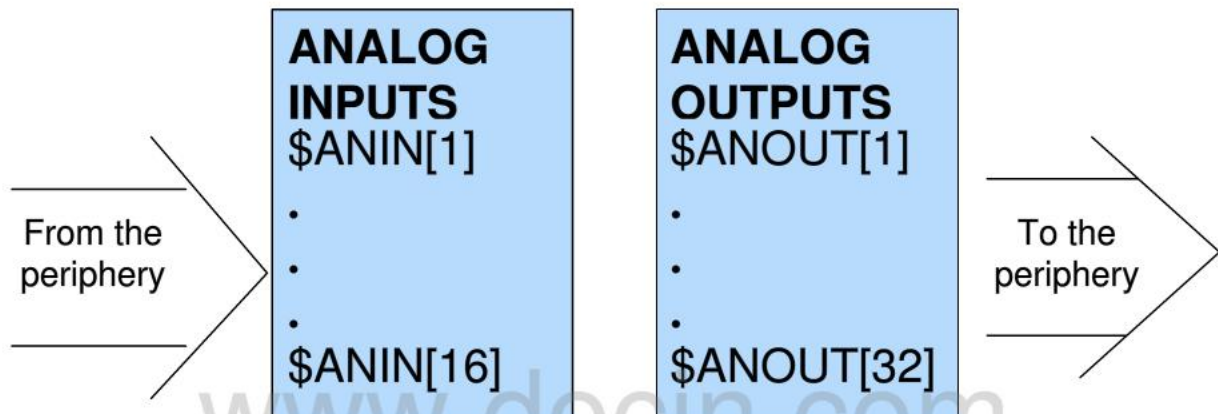
Example:

INB20=896,x8

OUTB20=896,x8



- An output on the PLC is an input on the robot.
- An input on the PLC is an output on the robot.





Robot side			Interbus side						
ANIN ANOUT	Index	=	Byte offset	,	Exponent 2	,	Type	,	Cal factor

Type : Justification, sign

- 0 : right, without sign
- 1 : right, with sign
- 2 : left, without sign
- 3 : left, with sign

CAL factor: Limitation of the value range

This entry is optional.

For the analog module, the specified value in digits corresponds to the nominal value (e.g. 10V).



Text page

www.docin.com

Example: Phoenix IB IL AI 2/SF:

IOSYS.INI:

From the manual:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VZ	AW														

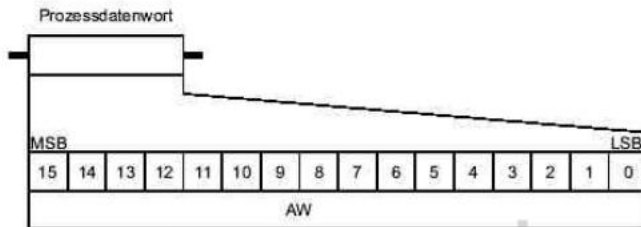
Ausgangsdaten- wort (Zweierkomple- ment)		-10 V bis +10 V U _{Ausgang}	Bemerkung
hex	dez	V	
<7FFF	32767	+10,837	
>7F00	32512	+10,837	
7F00	32512	+10,837	
7530	30000	+10,0	
0000	0	+2,667 mV	kleinster DAC-Quantisierungsschritt
0001	1	+333,33 µV	Prozessdatenauföösung
0000	0	0	
FFF8	-8	-2,667 mV	
8AD0	-30000	-10,0	
8100	-32512	-10,837	
<8100	gesonderte Behandlung:		
8001	-32767	+10,837	(Bereichsüberschreitung)
8080	-32640	-10,837	(Bereichsunterschreitung)
80xx	(sonstige)	letzten Wert halten	

ANIN1=4,16,3,CAL30000

Example: Phoenix IB IL AO 1/SF:

IOSYS.INI:

From the manual:



ANOUT1=4,16,2 or
ANOUT1=4,16,2,CAL65535

Prozessdaten-Ausgangswort OUT für den Spannungsausgang 0 V bis 10 V (Beispiel)																			
Spannungsausgang 0 V bis 10 V	Analogwert in Volt	Prozessdaten-Ausgangswort																	
		hex.	binär (Zweierkomplement)															LSB	
			MSB	15	14	13	12	11	10	9	8	7	6	5	4	3	2		1
10 V minus 1 QS	9.99985	FFFF	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10 V minus 2 QS	9.99990	FFFF	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
Halber ABE	5.0000	8000	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1 QS	0.153 mV	0001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Null	0.0000	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

FFFFHex = 65535Dec

Example: Phoenix ST modules:

IOSYS.INI:

From the manual:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VZ	AW												0/4	DB	BU

Eingangsdatenwort (Zweierkomplement)	0 mA bis 20 mA I_{Eingang}	0 V bis 10 V U_{Eingang}
hex	mA	V
7FF9	> 21,5	> 10,75
7FF8	20,0 bis 21,5	10,00 bis 10,75
7FF8	19,9951	9,9975
4000	10,0	5,0
0008	0,0048828	0,002441
0000	0	0

ANIN1=4,13,3 or
ANIN1=4,13,3,CAL4095

Note: 7FF8Hex = 1111 1111 1111 1000Bin

1111 1111 1111 1Bin = 8191Dec (13 bits)

With positive and negative range from – 4096 to + 4095

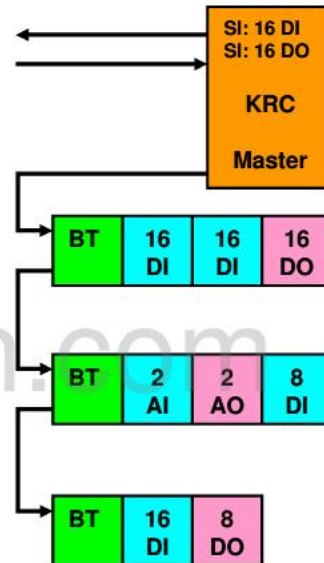
- **Slave ring:** 16 DI, 16 DO

- **Master ring:**

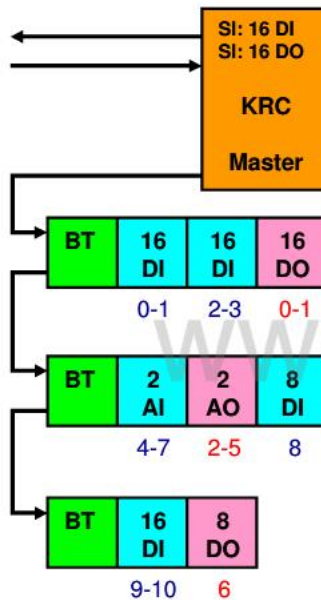
Bus terminal 1: 16 DI, 16 DI, 16 DO

Bus terminal 2: 2 AI, 2 AO, 8 DI

Bus terminal 3: 16 DI, 8 DO



IOSYS.INI

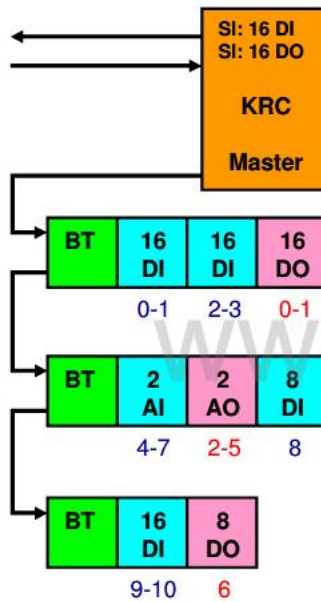


[INTERBUSPCI]

```
; Inputs of the master ring
INW0=0,x1           ; $IN[1-16]
INB2=2,x2           ; $IN[17-32]
ANIN1=4,16,3,CAL30000 ; $ANIN[1]
ANIN2=6,16,3,CAL130000 ; $ANIN[2]
INB4=8,x1           ; $IN[33-40]
INW5=9,x1           ; $IN[41-56]

; Outputs of the master ring
OUTW0=0,x1           ; $OUT[1-16]
ANOUT1=2,16,3,CAL30000 ; $ANOUT[1]
ANOUT2=4,16,3,CAL30000 ; $ANOUT[2]
OUTB2=6,x1           ; $OUT[17-24]
```


IOSYS.INI

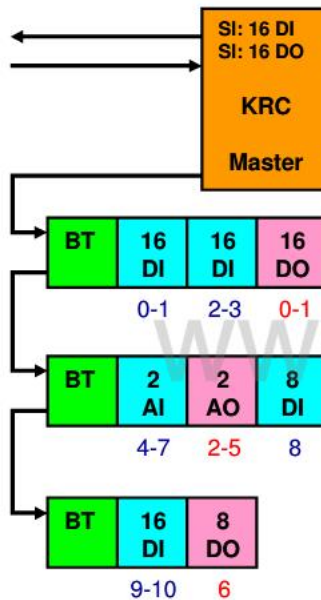


```
[INTERBUSPCI]
; Inputs of the master ring
INW0=0,x1          ; $IN[1-16]
INB2= ...
...
; Outputs of the master ring
OUTW0=0,1          ; $OUT[1-16]
ANOUT1= ...
...

; Inputs of the slave ring
INW12=896,x1       ; $IN[97-112]

; Outputs of the slave ring
OUTW12=896,x1      ; $OUT[97-112]
```

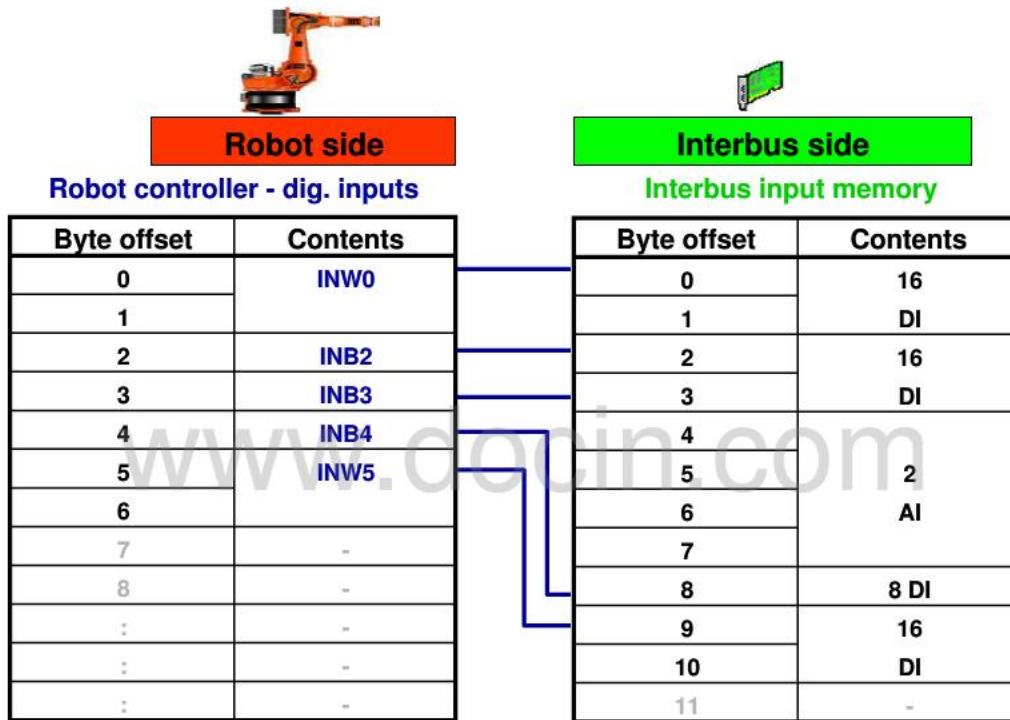
IOSYS.INI



```
[INTERBUSPCI]
; Inputs of the master ring
INW0=0,x1           ; $IN[1-16]
INB2=2,x2           ; $IN[17-32]
ANIN1=4,16,3,CAL30000 ; $ANIN[1]
ANIN2=6,16,3,CAL30000 ; $ANIN[2]
INB4=8,x1           ; $IN[33-40]
INW5=9,x1           ; $IN[41-56]

; Outputs of the master ring
OUTW0=0,x1           ; $OUT[1-16]
ANOUT1=2,16,3,CAL30000 ; $ANOUT[1]
ANOUT2=4,16,3,CAL30000 ; $ANOUT[2]
OUTB2=6,x1           ; $OUT[17-24]

; Inputs of the slave ring
INW12=896,x1         ; $IN[97-112]
; Outputs of the slave ring
OUTW12=896,x1        ; $OUT[97-112]
```





Robot side

Robot controller - dig. inputs

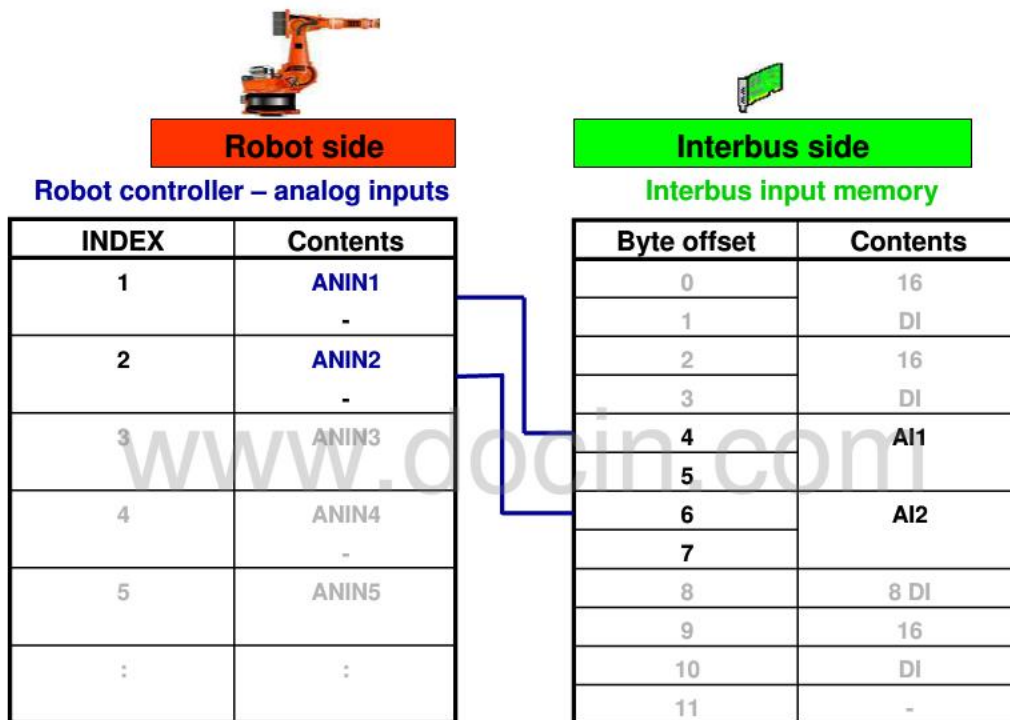
Byte offset	Contents
...	..
...	..
...	..
...	..
12	INW12
13	
...	..
...	..
...	..
...	..
...	..
...	..
...	..

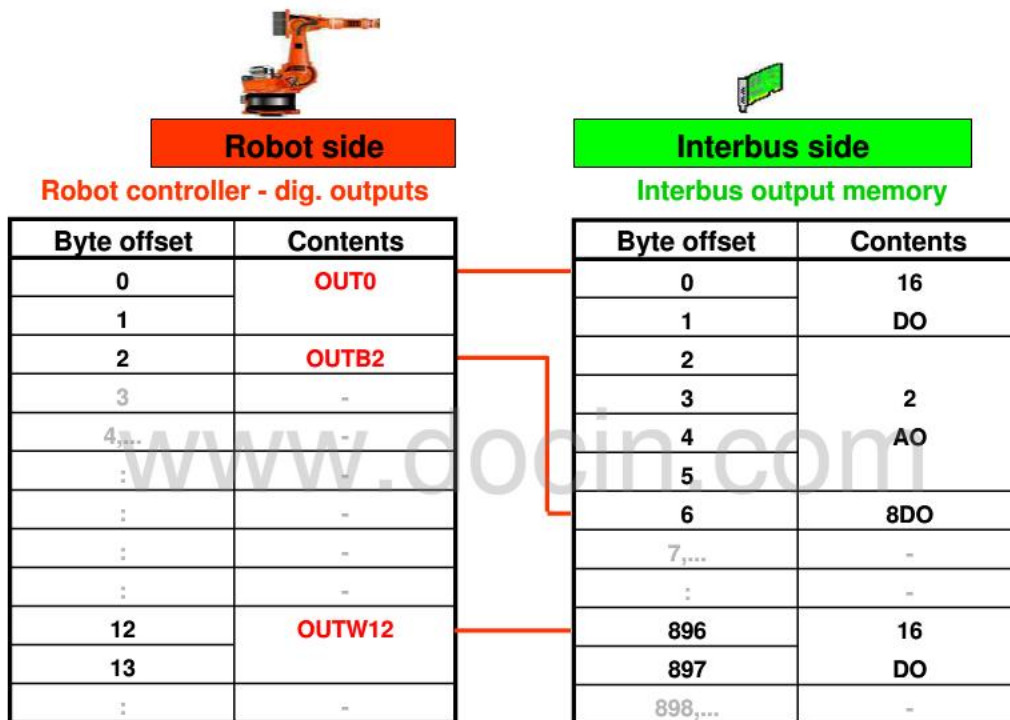


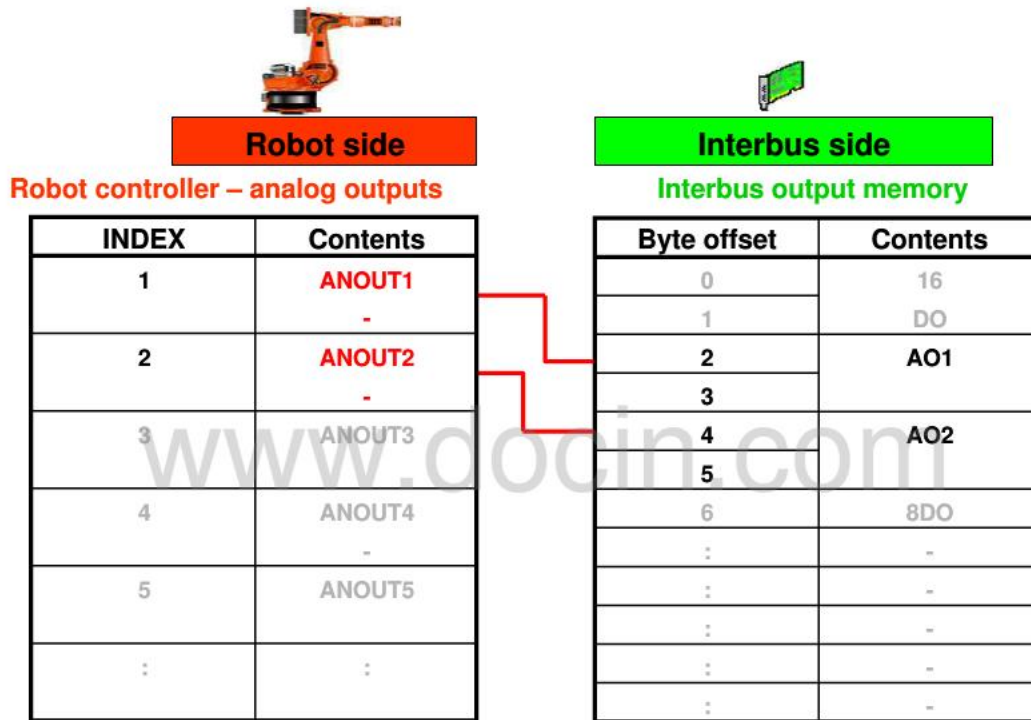
Interbus side

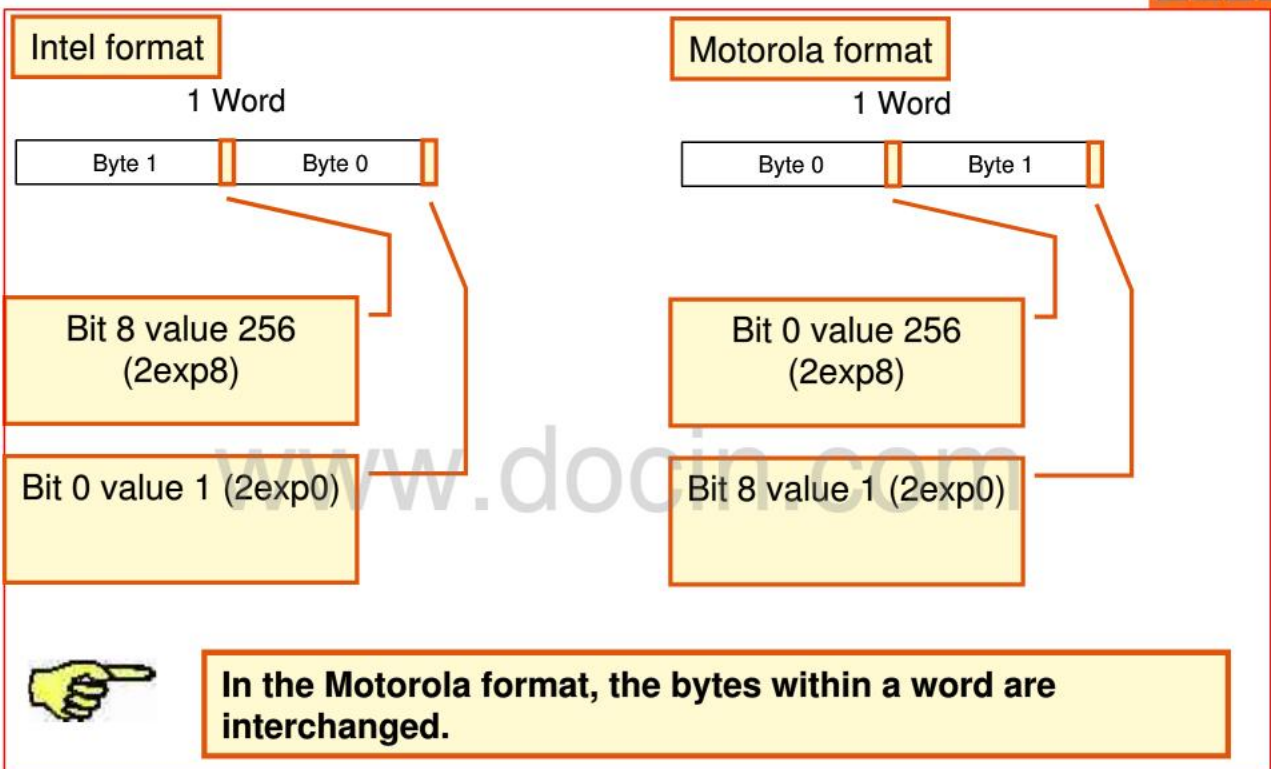
Interbus input memory

Byte offset	Contents
...	..
...	..
...	..
...	..
896	16
897	DI
...	..
...	..
...	..
...	..
...	..
...	..
...	..











PC-based controllers use the Intel format. (KUKA KRC)

Phoenix and Siemens use the Motorola format.



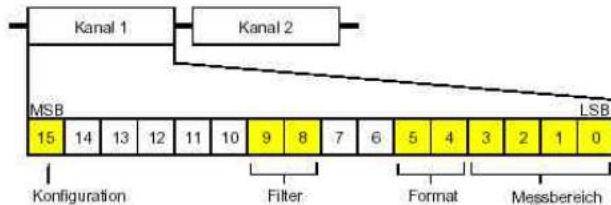
Depending on the module, it may be necessary to swap the bytes within a word!

Example:

IOSYS.INI:

```
INB0=1,x1 ;Word 0  
INB1=0,x1 ;Word 0  
INB2=3,x1 ;Word 2  
INB3=2,x1 ;Word 2
```

- There are modules (e.g. analog inputs) which can be configured by means of control signals (outputs on the robot).
- These control signals can be configured **if necessary**.
- Control signals can be used to specify the analog format, for example.



- There exist modules with 8 inputs, for example. These module occupy 1 work on the Interbus memory (Interbus memory), however. The unused byte has no function.

(Byte.Bit)- Sicht	Byte	Byte 0								Byte 1							
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Modul	Klemme (Eingangssignal)	1	2	3	4	5	6	7	8	nicht belegt							
	Klemme (24 V)	17	18	19	20	21	22	23	24								
	Klemme (Masse)	33	34	35	36	37	38	39	40								



There are modules (e.g. analog inputs and outputs) which always start at an even-numbered Interbus offset byte. The bytes in between are not used.

www.docin.com