



eScada

v24.2.0  
Drivers

eScada.Drivers.OmronCMode

## eScada.Drivers.OmronCMode

( Host Link Protocol – C-Mode commands )

### OS availability

Windows, Linux, RaspBian

### Atomic data type

16 bit Word oriented protocol.

### Hardware and documentation reference

www.omron.com

Specification W342-E1-16

### Parameters available in every section

Channel:	COM Port	Serial port name depending on OS type. e.g. Linux: /dev/ttyS0, /dev/ttyUSB0 e.g. Windows: COM1, COM3
	Baud rate	Communication baud-rate, eg. 9600, 38400, 19200, etc.
	Parity	N for none, E for even, O for odd
	Data bit	Allowed values are 5, 6, 7 and 8
	Data stop bit	Allowed values are 1 and 2
	Reconnect timeout [ms]	Waiting time before a reconnection after COMM break-down
	Response timeout [ms]	Timeout interval used to wait for a response.
Device:	Host link format	Host Link Formats available, 1:1 and 1:N (default)
	Unit ID	Unit ID number
	Read - Retry value.	Retry value before getting COMM error. (0=no retry)
	Write - Retry value.	Retry value before getting COMM error. (0=no retry)
Group:	none	
Tag:	none	

### Remarks for devices

The following attributes can be expressed for each device.

Bytes order actions	None, Swap bytes order, Swap bytes order in DWords, Swap words order, Swap bytes order in DWords then words order
String actions	None, Swap bytes in words

### Useful Linux commands

COM List:	<code>dmesg   grep tty</code>
COM rights:	<code>sudo chmod a+rw /dev/ttyUSB0</code>
COM user info:	<code>ls -l /dev/ttyUSB0</code>
COM add user:	<code>sudo adduser username dialout</code> - (dialout is the default group)

### Implemented memory area

CIO	<code>CIOx</code>	Read/Write	
LR	<code>Lx</code>	Read/Write	Treats CIO 1000 to CIO 1199 as a data link area)
HR	<code>Hx</code>	Read/Write	
Timers PV	<code>TPVx</code>	Read/Write	
Timers Status	<code>TSTx</code>	Read only	
Counters PV	<code>CPVx</code>	Read/Write	
Counters Status	<code>CSTx</code>	Read only	
DM	<code>Dx</code>	Read/Write	
AR	<code>Ax</code>	Read/Write	
EM	<code>Ee.x</code>	Read/Write	

x = Address

e = EM bank number

Expressed using base 10 numbers

### BCD values

If it's necessary to use BCD codification for integers values, please precede the value address with this character @. It is possible to use such kind of character with all integers format from 16 bits to 64 bits.

e.g.

@TPV3 (value using BCD format), TPV3 (value using binary format)

@D3 (value using BCD format), D3 (value using binary format)

## Addressing

Variable type	Type	Address type	Items
<b>Boolean</b> The number of items used declaring TAGs, must be a multiple of 16			
Single bit	Bit	CIO, L, H, D, A, E	(C)
<b>Byte</b> The number of items used declaring TAGs, must be a multiple of 2			
Unsigned 8 bit	UInt8	CIO, L, H, D, A, E	(C)
Signed 8 bit	Int8		
<b>16 bit</b>			
Unsigned integer 16 bit	UInt16	CIO, L, H, D, A, E, TPV, TST, CPV, CST	(C)
Signed integer 16 bit	Int16		
<b>32 bit</b>			
Unsigned integer 32 bit	UInt32	CIO, L, H, D, A, E	(C)
Signed integer 32 bit	Int32		
Single precision 32 bit - ( IEEE 754 )	Float		
<b>64 bit</b>			
Unsigned integer 64 bit	UInt64	CIO, L, H, D, A, E	(C)
Signed integer 64 bit	Int64		
Double precision 64 bit - ( IEEE 754 )	Double		
<b>Strings</b> The string length used declaring TAGs, must be a multiple of 2 String bytes can be interpreted as ASCII, UTF-7, UTF-8, UTF-16 or UTF-32 encoding			
Array of bytes	String	CIO, H, A, D	(A, C)
Array of bytes. (Siemens S7) Array of bytes. (AllenBradley style)	S7String ABString	CIO, H, A, D	(B, C)
(A) It depends on the strings length: e.g. if you want to read strings with a length of 10 chars each string, you can set a number of items of $74 / 10 = 7$ consecutive items.			
(B) It depends on the strings length: e.g. if you want to read strings with a length of 10 chars each string, you can set a number of items of $74 / (10+2) = 6$ consecutive items.			
(C) It depends on PLC model. The best way is to try with the maximum items you need. Please refer to your PLC model documentation and protocol specification W342-E1-16			

**S7 strings format**

They have got two bytes at the beginning.

The first byte is for max allowed string length, the second one is for the real string length.

These types of strings can be declared with a length of 255 bytes max.

**AB Strings format**

They have got one word (16 bit) at the beginning, it contains the string length.

**Consecutive items**

The number of consecutive read/write items depends on the PLC model.