

TUTORIAL KREO HMI Integrazione con Codesys

Tutorial dedicato alla integrazione tra KREO runtime ed il runtime Codesys 3.5

> Connect Ideas. Shape solutions.



# Introduzione

KREO supporta il driver CODESYS. La comunicazione in tal senso è gestita in 2 modalità e 2 driver diversi:

 CODESYS SOFT PLC (il driver per comunicare con il soft-plc CODESYS, sia esso quello integrato all'interno dei pannelli HMI sia quello per PC Control Win V3).

$+ \times \mathbb{C}$						
COM		ETH 1	ETH 2	ETH 3	ETH 4	Codesys
Info	)	Parameters	Codesys			
Proj	ject file					
Syn	nbol configura	ition file				
Pro	file					~
Ena	able automatic	import 🔽				
Last	t import symb	ols				
	Import Tag	<mark>15</mark>				

2. CODESYS SOFT PLC FOR GENERIC PLC (il driver per comunicare con un dispositivo con run time CODESYS sia esso un PLC fisico oppure un PC con a bordo CODESYS RTE).





L'unica differenza nei 2 drivers è nei parametri di comunicazione in quanto il primo richiede di indicare il .PROJECT-File, .XML-SymbolicFile, PROFILE TYPE del progetto CODESYS mentre il secondo ha delle impostazioni di parametri generici adattabili ai vari tipi di plc/RTE CODESYS 2.x, 3.x.



COM	ETH 1	ETH 2	ETH 3	ETH 4	Code
Codesys SoftPLC	(for generic PLC)				
					×
nfo	Parameters				
Device proper	ties				
Proper					
	n Dis	able Optimization			
User Address	n Dis Field Format Defau	able Optimization	~		
User Address	n Dis Field Format Defau	able Optimization	~		

# Device parameters

Download	Name	Value	Format
$\checkmark$	Instance	ESA_Codesys_Connection	
	Interface Type	Arti V2	
	Device	Tcp/Ip (Level 4): TCP/IP Level 4	
	Address		
$\checkmark$	Port	0	0 - 65535
✓	TargetID	0	0 - 65535
$\checkmark$	Motorola byteorder	Yes	
✓	Motorola	Yes	
	Gateway	Tcp/lp	
	Gateway address		
	Gateway port	0	0 - 65535
	Gateway password		
	NoLogin	1	0-1
$\checkmark$	Buffersize	0	0 - 65535
	PrecheckIdentity	0	0-1
			Ok



<u>Nota1</u>: Il firmware CODESYS cablato negli HMI-ESA ha profilo CODESYS 3.5SP6P7. Si richiede pertanto tale editor CODESYS nello sviluppare il progetto. Nessuna limitazione invece per quanto riguarda i progetti per PC in tal senso

### PREMESSA (files ESA per editor CODESYS)

Prima di iniziare il progetto occorre configurare l'editor CODESYS con alcuni files ESA che permetteranno di usare librerie e configurare dispositivi tipici di ESA, appunto.

Dal sito ESA si scaricano dunque i files in figura:

mation download $\times$ +												
C ŵ	O 🛔 https;	//www.esa-automation.	com/en/category-downle	oads/								
					Home	Company	Products	W	elcome,	customer.care	Besa-automat	ion.com FN 🗸
					Home	company	Troducts	Downoods	5109	conder us	Account	
		On this page you	can download all c	locuments about our pro		d certificati						
		codesys										
		SOFTWARE	ADD-ONS/DRIVERS	. LIBRARIES/TOOLS. SW VE	RSIONS, SE	RVICE PACKS	S & UPDATES					
		For downloads of	other software (V1	WIN, VTEDS or VTPROC	G) please s	send us a re	equest by c	licking here				
		Project Librar	y Archive	CODESYS								
		ESA CODESYS	library	CODESYS EW100AB								
				EW100AC EW600 I/O								
		CODESYS set	up 3.5 SP6 Patch 7	CODESYS EW100AB								
				EW100AC								

- <u>Project Library Archive</u>: progetto archivio da aprire su proprio PC. Questa operazione è sufficiente per installare su PC le librerie di sistema ESA necessarie per CODESYS.
- <u>ESA Codesys library</u>: files xml per la configurazione dei dispositivi ESA in CODESYS. Da installare in CODESYS>>TOOLS>>DEVICES REPOSITORY
- <u>Codesys setup 3.5sp6 Patch7</u>: setup dell'editor CODESYS richiesto per progetti HMI-ESA, come detto.



# Come fare:

1) Supponiamo di avere un progetto CODESYS con diversi tipi di tag (singole e strutture) nelle varie sezioni di progetto:

testVISU.project - CODESYS	
<u>File Edit View Project Deglarations Build Online</u>	<u>D</u> ebug <u>T</u> ools <u>W</u> indow <u>H</u> elp
管 📽 🖬   🎒   い つ 🐰 ங 🗈 🗙   🗛 🎼	ab   🏕 🕻 🌾 🌾   🛍   🛅 ▼ 📑   🏙   🎯 🧐 🕞 🔳 代   [≣ 5≣ 4⊒ 5≣ 6]   ⊅
Devices	
Device (CODESYS Control Win V3) Device (CODESYS Control Win V3) PLC Logic Application Struct1 (STRUCT) Struct2 (STRUCT) Struct2 (STRUCT) Struct2 (STRUCT) DEC PRG Device (CODESYS Configuration Symbol Configuration Task Configuration MainTask PLC_PRG	<pre>2 //Poly_1: ARRAY[09] OF struct1; 3 Poly_1: struct1; 4 Poly_2: struct2; 5 END_VAR</pre>
File Edit View Project Build Online Debu	ug Tools Window Help Ma ∰a la•b   ∕◆ 🎋 🎋   🛍    - 🗂      ௸ ௸ → 📄 ペ   Ç≣
Devices	
	<pre></pre>



testVISU.project - CODESYS

<u>File E</u> dit <u>V</u> iew <u>P</u> roject <u>B</u> uild <u>O</u> nline <u>D</u> ebug <u>T</u> ools <u>W</u> indow <u>H</u>	lelp			
🎽 🚔 🔚   🎒   い 🖙 🐰 ங 🛍 🗙   🏘 😘   a-b   🦽 🎋	** 3	<b>%</b>		≝ - 🖸   🕮   얮 🧐 → 🔳 🤏   💷
Devices 🗸	<b>џ</b>	×		🛛 Library Manager 🛛 🏲 📑 Symbol Configu
Image: Symplectic struct     Image: Symplectic struct struct     Image: Symplectic struct struct struct     Image: Symplectic struct s		•		1 TYPE struct2: 2 STRUCT 3 point21:REAL; 4 point22:DINT; 5 point23:STRING; 6 END_STRUCT 7 END_TYPE 8
testVISU.project - CODESYS				
<u>File E</u> dit <u>V</u> iew <u>P</u> roject <u>B</u> uild <u>O</u> nline <u>D</u> ebug <u>T</u> ools <u>W</u> indow <u>H</u> elp				
管 📽 🖬   🎒   🗠 🖂 階 🛍 🗙   🏘 🎼 arb   🦽 🎋 🎋 🌾		1	• <u>6</u> °	8 =* =* =? = <b>%   (</b> = •= •\$
Devices 👻 🖣 🗙		'n	Library M	lanager 📲 Symbol Configuration 🏹
		1 2 3 4 5 6 7 8	PRC	GRAM PLC_PRG pippo: INT; start: BOOL; i: UINT; scale: ARRAY [016] OF REAL; sec2: ARRAY [016] OF REAL; o_VAR

2) Seleziono nell'oggetto SYMBOL CONFIGURATION le tags che occorre importare nel progetto KREO.

CODESYS crea così un file .XML nella propria cartella di progetto (lo fa tramite comando BUILD>>GENERATE CODE oppure dopo un semplice download di progetto).



#### testVISU.project\* - CODESYS Ele Edit View Project Build Online Debug Tools Window Help 12112年1月1日(1日日日)(1日日日)(1日日日日)(1日日日日日)(1日日日日) ▼ X Ibrary Manager Symbol Configuration X S GVL PLC\_PRG Struct1 Device Struct2 View + W Build Settings + Devices testVISU Device (CODESYS Control Win V3) Changed symbol configuration will be transferred with the next download or online change E PLC Logic Members Comment Symbols Access Rights Maximal Attribute Type Syntoxis → ■ Constants → ♥ GVL → Ø Poly\_1 → Ø Poly\_2 → ■ IoConfig\_Globals struct2 (STRUCT) .... Poly\_1: ARRAY[0..9] OF struct1; 549 549 99 99 GVL struct1 struct2 Conary Manager PLC\_PRG (PRG) Symbol Configuration 🖹 📝 📄 PLC\_PRG Image: Constraint of the second se UINT Task Configuration INT 🚊 🥩 MainTask ARRAY [0..16] OF REAL ARRAY [0..16] OF REAL BOOL

3) Dalla sezione tags di KREO importo così il file XML creato da CODESYS.

File Edit	Action Tools Help		
TESTALLBRA2 ^			
	$\begin{tabular}{cccc} \hline $\nabla$ & Folder $\nabla$ & Name $\nabla$ & Description $\nabla$ & Type $\nabla$ \\ \hline $\nabla$ & Type $$	Address type 🖓	7 Provider ⊽
	_ Import tags		
Pages (1)	Import file selection		×
Popups (1)	← → ✓ ▲ → Ouesto PC → Disco locale (C:) → Sauro → temp	v ö 🖉 Cerc	a in temp
Sequences (1)			
	Organizza 🔻 Nuova cartella		≣≕ ▼ 🛄 😲
Alarms (0)	Questo PC Nome	Data	Tipo Dim
Datalogr (0)	Desktop demo_struct	11/06/2021 16:04	Cartella di file
	Documenti Documenti TestVISU Device Application xml	29/01/2021 17:40	Cartella di file
Recipes (0)	Download	00/01/2021 11/50	Documento Ame
Scripts (0)	E Immagini		
P <sup>™</sup> Javascripts (0)			
9 Javascripts external (0)	Video		
	Lisco locale (C:)		
() Timers	CCS (\\ITSHARE)		
Tasks (0)	📷 software (\\itfsp		
Pipelines (0)	🔀 Uts (\\ITSHARE\;		
	🗙 hw (\\ITSHARE\c		
🗘 settings 🗸 🗸	🙀 all (\\TSHARE) (		>
	Nome file: testVISU.Device.Application.xml	✓ Codesvs (	(*.xml) ~
		Apri	i Annulla
		Арп	Annuia

4) Le tags vengono così importate in KREO, siano esse strutture, array, singole tags e diventano disponibili per essere usate nel progetto.



KH 🖬 File Ed	lit Actio	on Tools Help					
TESTALLBRA2							
ل ۲۵ Configuration			Name ⊽	Descri	ption ⊽ Type ⊽	Address type	
	_/	Import tags					
Pages							
Popups	(1) Ta	ags to import					
Sequences	(1)	Name					
🖉 Tags 🥖							
	(0)			<b>-</b>			✓
曼 Datalogs	(0)			Folder	lag lype	Address lype	Description
Recipes	(0)				Structure	Device	
Cripts	(0)	Application.GVL.Poly_2		_	Structure	Device	
🖳 Javascripts	(0)				UnsignedInteger	Device	
🖳 Javascripts external	(0)	Application.PLC_PRG.pippo			Integer	Device	
() Timers	(0)				Real	Device	
💮 Tasks	(0)			_	Keal	Device	
Pipelines	(0)	Application.PLC_PRG.start			Boolean	Device	
🔍 SETTINGS	~						
			ewice Codesys :			Option Overnae if a	oject already exists V
					Can	cel	Vext 🕨

5) Nella procedura di download dei progetti KREO+CODESYS SOFT-PLC (embedded nel HMI-ESA) si puo' scegliere se trasferire i 2 progetti separatamente, ognuno con i rispettivi editor, in 2 operazioni quindi distinte oppure in un'unica operazione KREO.

Nella procedura di download occorre in questo caso selezionare o meno il check UPDATE CODESYS FILES in figura:





<u>Nota2</u>: Nel realizzare il progetto KREO si puo' utilizzare l'import automatico delle tag (vedi figura) ma attenzione che questo non porti a import indesiderati di tag-CODESYS nel progetto KREO.

Si consiglia dunque di disabilitarlo nel caso di un'operazione di import più controllata.

kH 🖬 File Ed	lit Action	Tools Help					
TESTALLBRA2		+ × C					
			СОМ				Codesys
Languages							
Dictionary						×	
ຼື =ຼິ Units		Info	Parameters C	odesys			
දි <sub>ධ</sub> Security users		Project file					
P Notifications		Symbol configuration fi	le				
		Profile				~	
		Enable automatic impo	rt 🗌				
		Last import symbols	06/07/2021 17:57:51				
		Import Tags					



<u>Nota3</u>: Nel realizzare il progetto CODESYS si consiglia di abilitare i 3 check in figura (right click su Application >> Properties) per mantenere allineato progetto e boot-app





**Nota4**: Nei progetti KREO con driver CODESYS SOFT PLC FOR GENERIC PLC abbiamo visto che i parametri di comunicazione generici sono i seguenti:

				×
Info		Parameters		
De	vice pro	perties		
	Soft	iscan Di	sable Optimization	
	User Add	ress Field Format Defau	it ~	
		-		
	Address p	refix		
De	vice par	motors		
De		lileters		
	Download	Name	Value	Format
		Instance	ESA_Codesys_Connection	
		Interface Type	Arti V2	
		Device	Tcp/lp (Level 4): TCP/IP Level	4
		Address	2	
		Port	0	0 - 65535
		TargetID	0	0 - 65535
		Motorola byteorder	Yes	
		Motorola	Yes	
	-	Gateway	Tcp/lp	
		Gateway address		
		Gateway port	0	0 - 65535
		Gateway password		
		NoLogin	1	0 - 1
	$\checkmark$	Buffersize	0	0 - 65535
		PrecheckIdentity	0	0 - 1
				Ok

Si vuole far notare che, in caso di progetti con sw basati su CODESYS 3.x, i parametri di comunicazione coinvolti sono quelli evidenziati in rosso (e non tutti necessariamente).



**Nota5**: Nella stessa figura il parametro ADDRESS PREFIX permette di importare tags-CODESYS eliminando il prefisso iniziale del nome.

Quindi per esempio, una tag con nome *Application.GVL.Poly\_2* e con ADDRESS PREFIX= *Application.* sarà importata in KREO con nome: *GVL.Poly\_2*.

**Nota6**: Nella stessa figura il parametro DISABLE OPTIMIZATION diventa utile in fase di debug.

Se abbiamo una situazione di 1 tag non definita nel plc si potrebbe avere a RT una serie di ??? su altre tag di pagina cmq configurate correttamente. Questo perché il protocollo di comunicazione ottimizza le richieste impacchettando nel frame inviato al plc il max. nr. di tags possibili. Questo però puo' inficiare la comunicazione su tutte le tag del pacchetto in caso di errore e cio' non aiuta l'utente a identificare quale tag causa il problema.

Il parametro DISABLE OPTIMIZATION forza invece una singola richiesta per ogni tag della pagina rallentando sì la comunicazione con il plc ma evidenziando esattamente i ??? sulle sole tag che presentano errore di comunicazione.



Connect ideas. shape solutions.

ESA S.p.A. | www.esa-automation.com |