# DADA OSOUTIONS Developing and exploiting machine intelligence

# **CODESYS Beginners Tutorial**

Getting started with CODESYS

Version: 1.3 Last updated: 15/05/2017 Author: Noor Mufti



# CODESYS Beginners Tutorial index

Introduction1
Objectives1
Prerequisites1
How to write your first program in XSoft – CODESYS 31
Step 1: Preparing the environment1
A) Open the development environment (IDE)1
B) Create a new Project1
C) Identify the project
D) Set the Project device and language3
Step 2: Write the program
A) Open the project
B) Select PLC_PRG (PRG)
C) Write the program
D) Name the Contact5
E) Add a coil to the program6
F) Name the coil
Step 3: Run this program8
A) Enter Simulation mode
B) Once you are in simulation mode8
C) Create the application on the device9
D) Run the simulation9
E) Manipulate the values
F) See how Button1 effects Bulb111
Conclusion

CODESYS Beginners Tutorial

Version: 1.3

Index



# Introduction

This tutorial is designed for aspiring programmers who wish to learn more about industrial and physical computing using the open source platform; CODESYS.

More information about CODESYS can be gained from the Smart Software Solutions (3S) website <u>www.codesys.com</u>.

# Objectives

In the course of this tutorial you will learn how to:

- Create and save new CODESYS projects
- Implement and control physical elements in a virtual environment
- Run virtual simulations to test your programme
- Have the confidence to further explore the programming environment

#### Prerequisites

There is no need for previously proven software development skills in order to successfully complete this tutorial. All that is required is a licenced (including a Demo licence) copy of CODESYS version 3.5 or higher, and basic computer literacy.

# How to write your first program in XSoft – CODESYS 3

# Step 1: Preparing the environment

#### A) Open the development environment (IDE)

Double click on the XSOFT icon to open the software.



#### B) Create a new Project

Within the homepage of the application select New Project...

CODESYS Beginners Tutorial Version: 1.3

> Page **1** of **11**

ces	+ 4 ×	🛃 Start Page 🗙		
	•	XSOFT-CODESYS V3.5.10		
	1			
		Basic Operations	Latest News	
		New Project		
		🎯 Open Project		
		Open Project from PLC	Cont	
		Recent Projects	-	

# C) Identify the project

In the New Project pop up select 'Standard Project', then an appropriate folder location and finally give your project a name.

Lib	raries	Templates:	
Pro	ojects	Empty project HMI	project Standard project w
A project o	notaining one device. o	project w	implementation for DIC DDC
A project c	ontaining one device, o	ne application, and an empty	Implementation for PLC_PRG
	myfirstproject		
Name:	-		~
Name: Location:	C: Users Documents		

CODESYS Beginners Tutorial

DAJO Solutions Developing and exploiting machine intelligence

Version: 1.3

Page 2 of 11



#### D) Set the Project device and language

Next, we need to select the hardware and the programming language to be used in the project. For the purpose of this tutorial we will use an Eaton XC152 PLC and Ladder Diagrams, although in the future thee options may vary depending on the application.

You are abou objects within	t to create a new standard project. This wizard will create the following n this project:	
- One program - A program F - A cyclic task - A reference	mmable device as specified below PLC_PRG in the language specified below c which calls PLC_PRG to the newest version of the Standard library currently installed.	
Device:	XC152 (Eaton Automation)	$\sim$
PLC_PRG in:	Ladder Logic Diagram (LD)	~

## Step 2: Write the program

#### A) Open the project

The project will open in an Integrated Development Environment. (IDE)

In this initial screen there are a number of key things to note. The menu bar on the left gives easy access to everything needed to create your program. From your earlier selections in 1D (Select the program device and language) it has remembered your preferences and has listed only the headings appropriate to this application.



CODESYS Beginners Tutorial

Version: 1.3

Page 3 of 11



#### B) Select PLC\_PRG (PRG)

PLC\_PRG(PRG) is the default name for all programs, you can easily update this by right clicking it, and then selecting 'Refactoring'.

B B B B B B B B B B B B B B B B B B B	4 (0) + = 4 (() () () () () () () () () () () () ()	Program Tools	
			$\frown$
	Program Variables	*	Sublex General Second Secon
(ID)	Program Construction	A weight	<ul> <li>Other Operators</li> <li>Function Macks</li> <li>Ladder elements</li> </ul>
			$\bigcirc$
a Statum		• + 4]	
a tagen - Total II arresto (1) it converges (1) manuages (1)			
		Lasibali 🧿 8 🕭 8 Pecanple: 🖌 Eur	et saler: (Hobody)

To develop the program there are three important areas. Firstly, the bottom middle section of the page where the program is constructed. Above this there is a section to detail the program variables. Lastly, to the right are the program tools which you will use to write the program.

#### C) Write the program

We are going to generate a simple program that turns on a light. In order to do this we firstly need a button.

From the toolbox on the right, select 'Ladder Elements' at the bottom of the list, this will open a sub menu of elements. From here drag and drop 'Contact' onto the 'Start Here' icon which will appear in the program construction area once you pick up the 'Contact' element.

15, <b>7</b> 86	Det lan	Bonch     Bonce     Bonce     Boolean Operators     Hath operators     Noter Operators     Nuction Nocks     Ladder elements     Defendent
		Vestat

CODESYS Beginners Tutorial

Version: 1.3

Page 4 of 11



an Angunet See Text State See Text State S	2 2 2 2 2	* Content     Entropy     * Content     Entropy     * Content     Entropy     * Other Society     * Other Society     * Other Society     * Other Society     * Entropy
		Constant Constant
	1914 <b>B</b>	Con     Constant Internation
	20 × <b>6</b>	file with Rights     we Argument     where     we files     we file     we files     we files     we files     we files     we file     we file
	(1814) <b>(1</b> 81	
* ∰ The Contractor * ∯ The Contractor * ∯ Art_Noi		A Junp     ore Raturn     A Juppt     Te Barch     South     South
* <b>€ verie</b> * <b>8</b> x∠se 1 	<b>6</b> .	4 Japit C Banch S Benzte 8 Boolean Operators • Roth operators • Roth operators • Roth operators • Instance Bools • Ladder elements • Ladder elements
	38 <b>K</b>	C Banch S Decute B Doolean Operators * Hath operators i Other Operators * Function blocks • Ladder elements Ladder elements
	100 % <b>(S</b> )	Boolean Operators     Boolean Operators     Holls operators     Other Operators     Cher Operators     Function blocks     Ladder elements     Bittenti
	30% <b>(S</b> )	Boolean Operators     Hath operators     Hath operators     Other Operators     Frunction Blocks     Ladder elements     Silvenets
	381% <b>(S</b> .	Hath operators     Other Operators     Trunction blocks     Ladder elements
	30 % <b>S</b>	Coher Operators     Function blocks     Ladder elements
		Ladder elements
		Not state
		45 Contect
		Negated contact
		Forallel contact
		M Parallel negated co
		e dal
		Sec con
		TON
		TOP TOP
		🗃 CTU
		e cm
		T MOVE
		-# Jenp
		T finish
		T Branch Rantified
	N ALC HARRY	6
2 Deven	* + - 10 % B	A 4
Research and the second		

#### D) Name the Contact

Without a name, CODESYS callsany element '???'. Clicking this will allow you to replace it with something appropriate to the application. When you press return a dialogue box will open with further details of the element. Press OK.

Buttoni.	Scope: Name: VAR V Button1	Type: BOOL V >
	Object: Initialization: PLC_PRG [Applcation]	Address:
	Flags: Comment:	
	RETAIN PERSISTENT	0
		OK Cancel

CODESYS Beginners Tutorial

DA DAJO Solutions Developing and exploiting machine intelligence

Version: 1.3

Page 5 of 11

You will also see at this point that the variables for the button have automatically been added to the program variables section.

	·告告• · · · · · · · · · · · · · · · · · ·
/	
	2 VAR
	3 Button1: BOOL:
	4 END VAR
	1 Descent
	BULLORI

#### E) Add a coil to the program

Just as we need an input to activate the program, we need an output to do something once activation has occurred. From the Ladder Elements menu drag and drop a 'Coil' element onto the end of the Contact, 'Button1'. Just like last time, as soon as you pick up the coil the program will react.

	* Hath operators	
A 7	Instrumt Products     Instrumt Products       Instrumt Products     Ladder elements       Instrumt Products     Image: Image Products       Instrumt Products     Image Products       Instrumt Products <td>e</td>	e
	W Parallel context W Parallel context W Parallel inspired (* > Call (*) Statul.,	f con
	↔ ( <u>Acol</u> ) 38 TOM 38 TOM 38 TOM 38 TOM	
	er co 9 Mote Jung - Raturn T, Sanch	
	T Bandi Sarybi	

When the coil is dropped into place correctly the program will look like this:

RC_PRG x	+ Toollo	r .
i PRODAW RC_NB VAR Batemii BOKr EB0_NA	■ cm ■ * 50	eeral Network Sox Sox Sox Sox Sox Sox Sox Sox
	100 % (R) + OI	er Operators Inction blocks
Betteni		Network Contact Negated conta
		Parallel contect     Parallel negate     Parallel negate     Ceil     Set coil     Reset coil     TON     TON
		CTU CTD MOVE

CODESYS Beginners Tutorial

**DAIO** Solutions

Version: 1.3

Page 6 of 11

# DA DAJO Solutions Developing and exploiting machine intelligence

# F) Name the coil

£

Auto Declare				×	Bulbi
Scope: VAR Object: PLC_PRG [Application]	~	Name: Bulb 1 Initialization:	Type: BOOL Address:	~ >	
Flags: CONSTANT RETAIN PERSISTENT		Comment:		Ç	

The program will automatically update all the relevant sections and will now look like this:

Mar Society on Mills 名称 Article 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2			
PLC_PRG x         Improve FLC_PRG           1         Noncone FLC_PRG           2         Build_1: BOOL:           3         Build_1: BOOL:           4         Build_1: BOOL:           5         EB_VAR		Toollos  → Junp → Junp → Hatun → Junp wer Retun 4 Juput T Brach 5 Decole + Bala operators + Hath operators + Hath operators + Other Opera	<b>P</b>
<b>▼</b> + Q. m	× 🖪	Reset coll     B TON     B TOF     CTU     CTU	

CODESYS Beginners Tutorial

Version: 1.3

Page 7 of 11

# DA DAJO Solutions Developing and exploiting machine intelligence

## Step 3: Run this program

#### A) Enter Simulation mode

Without a PLC connected, you need to run the program in simulation mode. Click on Online, then Simulation to turn on simulation mode

i) 🚅 ( 🧐	Login Alt+F8	▲ 🎋 🎋 🌾   ╚   🛅 • 👔   盥  ◎\$ 🧐 → 📲 🔍 (፲= ?) 🛀 *
- (c) (Q	Logout Ctrl+F8	[發發發→ ◎ @ 単] ◎ ◎       七張発力
evices	Create boot application	PLC_PRG X
my	Download	1 PROGRAM PLC_PRG
<u> </u>	Online Change	E 2 VAR
÷.	Source download to connected device	4 Bulb1: BOOL;
	Multiple Download	5 END_VAR
	Reset warm	
	Res <u>e</u> t cold Ctrl+Shift+R	
	Rese <u>t</u> origin	
	Simulation	
	Security +	
	Operating Mode	
		1 Button1

#### B) Once you are in simulation mode

Ensure that SIMULATION appears with a red highlight in the bottom right hand corner of the page. Then run the program by selecting Online, then Login.

P 19	Logn Att+P8	★ ≱ ≱   例  圓 • 0,1 圖  截 改 ♪ ■ 水(白 河 河 河 河 )  約  単	
64 09	Create boot application	1999号·日金玉山田田田一下升的方	
	Queriload Online Chagge Gource download to connected device Builgeb Counsiad Ganet warm Reset onlig Serulation Serulation	>         P ■ PRC/PRG ×         •           1         FROGUM FLC_PRG         ●           3         Sutton1: ROGL;         ●           4         Builbi: ROGL;         ●           5         Sutton1: ROGL;         ●           6         Builbi: ROGL;         ●           1         Button1         ●	Tooblox
POUs 🔮 🛙	angentry Ogenating Mode	(*) (*) (*) (*) (*) (*) (*) (*) (*) (*)	Bookean operators     Heth operators     Heth operators     Other Operators     Hother Operators     Hotcols     Lodder elements     Centact     W Parallel

CODESYS Beginners Tutorial

Version: 1.3

Page 8 of 11



# C) Create the application on the device

XSOFT-C	ODESYS-3 - PLC	programming syste	em (Demo license)	×
?	Application 'Sin want to create	n.Device.Application' of a state of the second s	does not exist on dev download?	ice 'Device'. Do you
		Yes	No	Details

Download the program to simulation mode by selecting Yes in the dialogue box.

# D) Run the simulation.

Click debug, then Start

<ul> <li>S wythetproved</li> </ul>		Manufacture & Annalisation and							
<ul> <li>A S Device Remedie</li> <li>B PLC Logic</li> </ul>	Tagle Draipart	pion 19		Type	Value	Prepared value	Address	Comment	
= O Application	O Quality Breakport	Bulb 1		800.	FALSE				
PLC_PRC	B Dable Drugsport	11							
= 10 Task Cor = 10 Hart	i Sapger /	De De	ttoni						(Bulki)
10	BagOA SHA								
	12 But to Curser								
	Set regt Statement	-							
Service and Services	The same of the second statement								+ + Q 100
Sector 2	Error values	1							
lupression	Unforce values Alt-		Apples	tion Type	Value	Prepared value	Execution point		Address Comm.
	Plan Cantral								
	Cave Dugp								
Products 1 (2) Stradiustics 3 Meanager - Total (Lamoda)	S-unning(s), S-messape(s) Laurholdt 🔥 O. 🔶 O. – S			-	an In align	Berry	ad going f		8 age Suda A
2 mart 1 20 tradicante 2 Managar - Tablé Lamardo)	(vartig(i), 1 matapol) (vartial: 0, 0, 0, 1			- For	ee headad	_	a di gang d		a age 5-10-10 <sup>1</sup>
Press 1 20 Bradquerin B Messagar - Total Berrar(c)	Praeming(s), 1 messapels			- For	ee Tranfait				er vager Facilitada (
Pinath 1 🕉 Insteams B. Heavyer - Stat Levelds	Paering(s), I messapels			-	ee boadgit			- Area	er veger Fusikadad
Presson 1 🐼 Bradqueria B. Messager - Tatol 2 area(cl)	Suanting(b), Smassayd(), Suanting(b), Smassayd(), Suanting(b), Smassayd(),			-	en tradiçă	har	riest	- Aug	a ungi Kubuduk
Pressper-Tool Ferrica	turbat on a	and a		-	an haafad		chpost		
Press 1 (2) Broducetts 5 Hearague - Total Entre(1)					en handad		erk poset		r oge foderadet
Presspor - Total Beneficia	Duarting(), 1 messape() tarbials 0 0 0 0		),		a hadad				a naga Radinak I
P math 1 🐼 Brokenth B. Messager - Total Enrolds	Sventing(s), S messapoli Inaribadi A A A A A		),	-	an tradict				a vaga finikasika
Presson 1 20 Branquero B. Messager - Total 2 arendol			),		en haded		-toost		
Press 1 B brokunts B Trensper - Staf I arre(5)	Duarting(s), 1 message(s) tankadi (k to k to s)		2						a rage Roberto

CODESYS Beginners Tutorial

Version: 1.3

Page 9 of 11



#### E) Manipulate the values

Once the program is running in simulation mode, you can simulate the inputs and outputs. To do this, click on the prepared values cell of the element you want to manipulate. This will toggle it through the possible options for that element. In this case, TRUE, FALSE, or blank.

Device Application.PLC_PRG						
Expression	Туре	Value	Prepared value	Address	Comment	
Button1	BOOL	FALSE	TRUE			l
Bulb1	BOOL	FALSE				
Buttop1 <true></true>						Bulb1
Buttop1						Bulb1

Try Clicking in the "prepared value" column in front of the "Button\_1" tag. this will change the value to "TRUE". Now write the value by using menu Debug, then Write Value.

ect	FBD/L	D/IL Build On	line <u>D</u> ebu	g <u>T</u> ools
CH.	Þ	Start	F5	5 76
00-1		Stop	Shift+F8	1 1
		Single Cycle	Ctrl+F5	PLC P
	10	New Breakpoint		ice.Ap
necte	5	Edit Breakpoint.		sion
		Toggle Breakpoi	nt F9	Buttor
cation	0	Disable Breakpoi	int	Bulb1
C PR	0	Enable Breakpoi	nt	
isk Con	Ç.	Step Over	F10	
Main	φ <sub>Ξ</sub>	Step Into	F8	
æ	¢.	Step Out	Shift+F10	
	*3	Run to Cursor		F
	\$	Set ne <u>x</u> t Statem	ent	
	ф	Show next State	ement	
		Write values	Ctrl+F7	
		Force values	F7	
		Unforce values	Alt+F7	
	T	Flow Control		
		Core Dump		•
		Display Mode		•

CODESYS Beginners Tutorial

Version: 1.3

Page 10 of 11

#### F) See how Button1 effects Bulb1



Turning Button1 on or off will have a corresponding effect on Bulb1. However, writing values to Bulb1 has no impact on either the button or the bbulb element. So when the "Button\_1" contact is ON, it turns ON the "bulb\_1" coil.

# Conclusion

Obviously, using a PLC to turn a lightbulb on or off would in many instances be considered to be an over engineered solution. But it does help answer the important question:

#### How many software engineers does it take to change a lightbulb?

This little project does capture some of the essential elements of developing a PLC program for a physical or even industrial application and we will build on this in our next tutorial: Using COSESYS to program an industrial device.

CODESYS Beginners Tutorial

AlO Soluti

Version: 1.3

Page 11 of 11