Error Checking with "POU for implicit checks"

Sometimes it happens that the application on the controller crashes mysteriously with a message such as "Access violation". Double-clicking the "Source position" in the PLC log jumps to the code position, showing crash positions that do not appear to have any connection to the problem (for example, a simple TON box or a library). Another indication for these types of errors is when the code position changes when the application is changed.

The common cause is writing beyond the bounds of an array.

This can be detected, for example, by using "POU for implicit checks", and here particularly with the function "CheckBounds".



Enampior

Requirement

- Create a "Standard project" and select "CODESYS ControlWin V3" as the device.
- Define the target system by means of the Network scan.



• Adapt the "PLC_PRG" POU as follows:

Declaration
VAR
<pre>iIndex : INT; astInst : ARRAY [15] OF INT := [5 (101)]; xActivate : BOOL;</pre>
END_VAR



Downloading and starting the project

- Download the project to the controller and start the application.The following image is displayed after you set the xActivate variable:

	Expression	Type	Value
	iIndex	INT	235
			200
Library Manager	astInst[1]	AKKAT [15] OF INT	101
E PLC_PRG (PRG)	astinit[1]	INT	101
Task Configuration	astinist[2]	INT	101
🖃 🥸 MainTask		INI	101
PLC_PRG	astinst[4]		101
	astinst[5]	INT	101
	I IF xActivate FAL	SE THEN	
	2 XActivate F	ALSE := FALSE;	
	□ 3 ● FOR iIndex	235 := 0 TO 5 DO	
	4 astInst	[11ndex 235] ???	:= 234;
	S END_FOR		
	G G END_IPRETORN		

ор (iIndex has value "0") because it is located in the memory area before the array.

In the next executed loop, it is detected that the cancellation condition of the loop is fulfilled, and as a result no element of the array will be written.

These kinds of mistakes in the memory can have unexpected results, as severe are the controller crashing.

Including the function "CheckBounds"

(!)

• Add the object "POU for implicit checks" to the project:



The following dialog opens automatically, where the type of check can be selected. Select the option Bound checks.

Add POU for implicit checks
Create special check functions for an application (for checking array bounds, divisions and pointer)
Available functions:
✓ Bound Checks
Division Checks
Range Checks
LRange Checks
Pointer Checks
Note: Adding a checkfunction will provoke a full recompile and prohibit an online change
Add Cancel

Using the function "CheckBounds"

- Download the project to the controller and start the application.Set a breakpoint at the desired check.

Only one option should be selected. The check may have to be repeated with another option.

If a bound violation is detected, then the project is halted at the breakpoint.

Devices v # x	PLC_PRG	unds 🗙 📆 Device					-
- D CheckBoundsExample	Device.Application.CheckBo	unds					
Image: Solution of Science (Connected) (CODESYS Control Win V3) Image: Solution (run) Image: Solution (run)	Device Application.CheckBo Expression CheckBounds burger Line and a second seco	unds Type DUNT DUNT DUNT DUNT DUNT DUNT DUNT DUNT TYP TYP TYP Type	Value <set breakpoint="" in="" order="" this="" to="" variable="" watch=""> <set breakpoint="" in="" order="" this="" to="" variable="" watch=""> <set breakpoint="" in="" order="" this="" to="" variable="" watch=""> Set Breakpoint in order to watch this variable> III elt sich hierbei un einen Implementier EN THEN 2 7 ; t su setsen, Meldungen aufsuseichnen, and SysTypes2_Itf as nevest. III</set></set></set>	Prepared value	Address	Comment	₩ ₩
	Messages - Total 0 error(s), 0 warning(s), 0 message(s)						7 X
	 O error(s) O warning(s) O message(s) X 						
	Description		Proj	ect Obje	ct Pi	osition	
POUs 💥 Devices	Messages - Total 0 error(s), 0 wa	rning(s), 0 message(s) 🚂 Watch	h 1 🖓 Breakpoints 🙀 Call Stack				
Last build: O 0 • 0 Precompile: V	RUN	Program loaded	Program unchanged Current user	: (nobody) II	VS Ln 34 G	ol1 Ch1 🕅	0

• Switch to the PLC_PRG POU and set the xActivate variable to TRUE.

The projects stops at the breakpoint regarding lower bound violation.



• Exit the CheckBounds function by stepping (press F10 two times) and the debugger jump automatically to the position where the boundary violation was detected:

Devices 👻 🕫 🗙	PLC_PRG x D CheckBou	inds 💮 Device						•
CheckBoundsExample	Device.Application.PLC_PRG							
🖹 😔 🛐 Device [connected] (CODESYS Control Win V3)	Expression	Tupe	Value	Preparer	dualue Ade	trace Co	mment 1	
E DI PLC Logic	Expression	Type	value	Preparec	a value Add	iress Co	nment I	۴I
Application [halt on breakpoint]	tindex	ADDAY [1 ELOF INT	0					
💼 Library Manager	* Vastinst	ARRAT [15] OF INT	EN CE					
PLC_PRG (PRG)	XACTIVALE	BOOL	PALSE			_		
CheckBounds (FUN)	·							
Task Configuration	I I IF xActivate FALSE	THEN						
🗏 🥵 MainTask	2 xActivate	FALSE;						
· ● PLC_PRG	FOR 1Index	0 := 0 TO 5 DO	-= 234.					
	5 END FOR 6 END_IPEEURH							
	Married Table and A. Sumah						100 %	<u> </u>
	Messages - Total O error(s), O warnin	ig(s), 0 message(s)					* 0	~
		• Uerrol	r(s) v warning(s) v message(s	^ %				-
	Description			Project	Object	Position	1	
POUs Z Devices	Messages - Total 0 error(s), 0 war	ming(s), 0 message(s) 🐺 V	Watch 1 🗃 Breakpoints 🖗 Call Stack					
Last build: 🔷 0 🔮 0 Precomple: 🗸	HALT ON BP	Program loaded	Program unchanged	Current user: (nobody)	INS	Ln 4 Col 9 C	h9 🕅	

• As an alternative, the call stack can also be displayed and and from there a jump made to the relevant position:

Devices 👻 🕈 🛪	PLC_PRG CheckBou	inds 🗙 👩 De	vice						-
CheckBoundsExample	Device.Application.CheckBou	nds							
O Device [connected] (CODESYS Control Win V3)	Expression	Туре	Value			Prepared value	Address	Comment	
- PLC Logic	V CheckBounds	DINT	0						
 O Application [halt on breakpoint] 	🍫 index	DINT	0						
Library Manager	🍫 lower	DINT	1						
PLC_PRG (PRG)	🍫 upper	DINT	5						
Checkbounds (FUN)									
Task Configuration	•			A 7					·
= S ManTask	1 // Automatisch er	zeugter Coo	ie: Es handelt sich i	hierbei um ein	en Implementierun	gsvorschlag.			^
- 몐 PLC_PRG	3 CheckBounda	< lower	lower 1 h						=
	a 4 ELSIF index) > upp	r 5 THEN						-
	5 CheckBounds	0 :=	upper 5 ;						
	B 6 ELSE	0	index 0						
	8 END IF		index 0 ;						
	9 -								
	10 (*Es ist auch mög	lich, einer	Haltepunkt zu setze	en, Meldungen	aufzuzeichnen, oo	ler beispiels	veise an e	iner Except	tic
	11 Add CmpApp. 11bras	y, SysExcep	ot.library and SysTyp	pes2_itf as no	evest.			100 %	<u>-</u>
									•
	Call Stack							•	ά×
	Application: Device.Application 1	Task: MAINTASK							_
	POU		Location	Instance path					
	 CheckBounds [Device: PLC Logi 	c: Application]	Line 3, Column 2 (Impl)						
	PLC_PRG [Device: PLC Logic: A	pplication]	Line 4, Column 1 (Impl)						
POUs Z Devices	Messages - Total 0 error(s), 0 war	ning(s), 0 messa	ge(s) 🔛 Watch 1 🗟 Break	points 🚯 Call Stack	t				
Last buk	d: 🔍 0 🖲 0 🔹 Precomple: 🗸	HALT ON E	Program loade	ed	Program unchanged	Cu	rrent user: (no	body)	Ø