ETAS

How To: RTPC Debugging



Symptoms:

Simulation stops and model gets automatically disconnected from the Target Server with a "SIGSEGV" error message in EE Application Log Window.



Reason:

After inspection of the RTPC log files a "SIGSEGV" (segment violation) error during simulation was seen.

That means there is an illegal pointer access within source code.

To investigate which is the cause of the error, the RTPC offers some kind of debugging possibilities which can be used to identify and repair (if possible) the root cause of a model crash.

Please follow the next guidelines:

1.) Stop the Simulation Controller.

2.) Select Configure LabCar-RTPC from the Main Page of the LABCAR-RTPC-Web-Interface

3.) Go to the LabCar-RTPC Configuration and choose the following settings for debugging purposes:

- RTPC_LOG_LEVEL = debug
- RTPC_COMPILE_OPTIMATION = 0
- RTPC_COMPILE_LINK_DEBUG = yes

Press Button Save LABCAR-RTPC Configuration:

e	Size of the trace buffer (in traceable events). Help	
	RTPC_COMPILE_OPTIMIZATION	0 🗸
	Optimization level of the compiler. Help	
	RTPC_COMPILE_LINK_DEBUG	yes 💌
	Place debugging information in generated code. Help	

4.) Build the whole LabCar-Project again via Build LCO Project.

5.) Start the Simulation Controller from the Main Page of the LABCAR-RTPC- Web-Interface.

6.) Open and Start the newly built Experiment.

7.) Wait until the SIGSEGV error occurs and note the address after catched at in the Application Log Info:

	📋 Application Log 🤝 🗸 🖉							
	Time	Component	Message		-			
i	09:32:04	TargetServer Device	Transfer real time model to RTPC 'RTPC' with IP Address '192.168.40.14' completed.					
i	09:32:04	LabCar Signal Source	Transferring real time models to targets completed.					
i	09:32:05	LabCar Signal Source	Model Download completed successfully.		Π			
i	09:32:06	TargetServer Access	Waiting until Models Init Task has been completed on all RTPCs.					
i	09:32:07	TargetServer Access	Init Task has been completed.					
i	09:32:08	LabCar Signal Source	Simulation started.					
1	09:32:08	RTPC Syslog	**Signal SIGSEGV caught at [0x0x8065ef8] (pid:10671/10716, si_pid:0, addr:0, code:1-Address not mapped). Stop execution (This log message may appear again)		1			
1	09:32:08	RTPC Syslog	Signal SIGSEGV caught at [0x0x8065ef8] (pid:10671/10716, si_pid:0, addr:(nil), code:1-Address not mapped). Stop execution. Registers: eax:00000000 ebx:b7d572					
i	09:32:14	LabCar Signal Source	Disconnect complete. Targets have been removed					
1	09:32:16	LabCar Signal Source	Lost connection to Target or Simulation Controller has been stopped. You have to download your model to the RTPC again.					
					-			
Application Log 🐞 Hardware Output								
		9	Disconnected Stopped Script recorder idle	DefaultExp				

8.) Select System Info from the Main Page of the LABCAR-RTPC-Web-Interface

9.) Go to the Debug / Disassemble section and Press Button Analyze:

Debug / Disassem	ble				
Additional debug information about the downloaded simulation model can be provided by analyzing the binary. The output will be printed into a new browser window. It is highly recommended to generate debug information into the executable as this eases typically the identification of errors.					
Model/VECU:	LABCAR model 💌				
Requested information: GDB Core Dump Listing 👻					
Analyze					

10.) Search for the noted SIGSEGV error address to debug your project, cf. 7., for my example 0x08065ef8:

disassemble /m \$pc Dump of assembler code for function cmod_Process_Controller_Controller:							
57	{						
	UX08065e10	: <+u>:	pusn	seop			
	0x08065e1d	1 <+1>:	mov	sesp, sebp			
	0x08065e1f	E <+3>:	sub	\$0x28, %esp			
38		// Get Inport	53				
39		<pre>#include "cor</pre>	ntroller	copyinports.h"			
40							
41		// Enter your	r code he	ere:			
42		// Make	the Poi	nter to bring to a SIGSEV during Simulation			
43		static :	int *poi	nter = NULL;			
44		static :	int coun	τ;			
45		if (cour	nt++ > 2	00)			
	0x08065eda	a <+190>:	mov	0x80dd4ec,%eax			
	0x08065edf	<+195>:	cmp	\$0xc8, %eax			
	0x08065ee4	<+200>:	setq	\$d1			
	0x08065ee7	/ <+203>:	add	\$0x1, \$eax			
	0x08065eea	a <+206>:	mov	Seax. 0x80dd4ec			
	0x08065eef	<+211>:	test	\$d1.\$d1			
	0x08065ef1	<+213>:	ie	UX8065efe < mod Process Controller Controller+226>			
46			1				
47			1	topinter - 10. // Hier kracht es			
- '	00006565	2 212155	mott	ovolddifo Say			
	0x00065cf	2+2205.	morrl				
~	OX00005E10	NT2202.	HUVL	φυλα, (stax)			
40			1				
40			1				
49							
50							
51		//					
52		//werte	einiese	n:			
53		//					
54		interna.	1_Curren	t_Floor=Inport_Current_Floor;			
	0x08065efe	<+226>:	fldl	0X80dd18			
	0x08065f04	<+232>:	fisttp	1 -0xc(%ebp)			
	0x08065f07	<+235>:	mov	-Oxc(%ebp),%eax			
	0x08065f0a	a <+238>:	mov	%eax, 0x80dd4d4			



With the correct optimization settings you should be able to see in which module at which line of the code the model has crashed.

The generated report at the RTPC with the debugging results (usually called by default **ETAS RTPC.mht**) contains the information about the code.

Please do not forget to undo the compile settings mentioned above, after your analysis is complete.

i Additional information:

This procedure was tested with LABCAR-OPERATOR 5.3.0 and RTPC 6.1.0.

In case of further questions:

You will find further FAQs here: www.etas.com/en/faq

Please feel free to contact our Support Center, if you have further questions.

Here you can find all information: http://www.etas.com/en/hotlines.php

This information (here referred to as "FAQ") are provided without any (express or implied) warranty, guarantee or commitment regarding completeness or accuracy. Except in cases of willful damage, ETAS shall not be liable for losses and damages which may occur or result from the use of this information (including indirect, special or consequential damages).