Multicore Example,

Use of Address Range Qualification with MCDS TraceViewer And Lauterbach

Robert Valascho

Multicore Example

Simple Multicore Example is developed using AURIX^{TM} Developer Studio

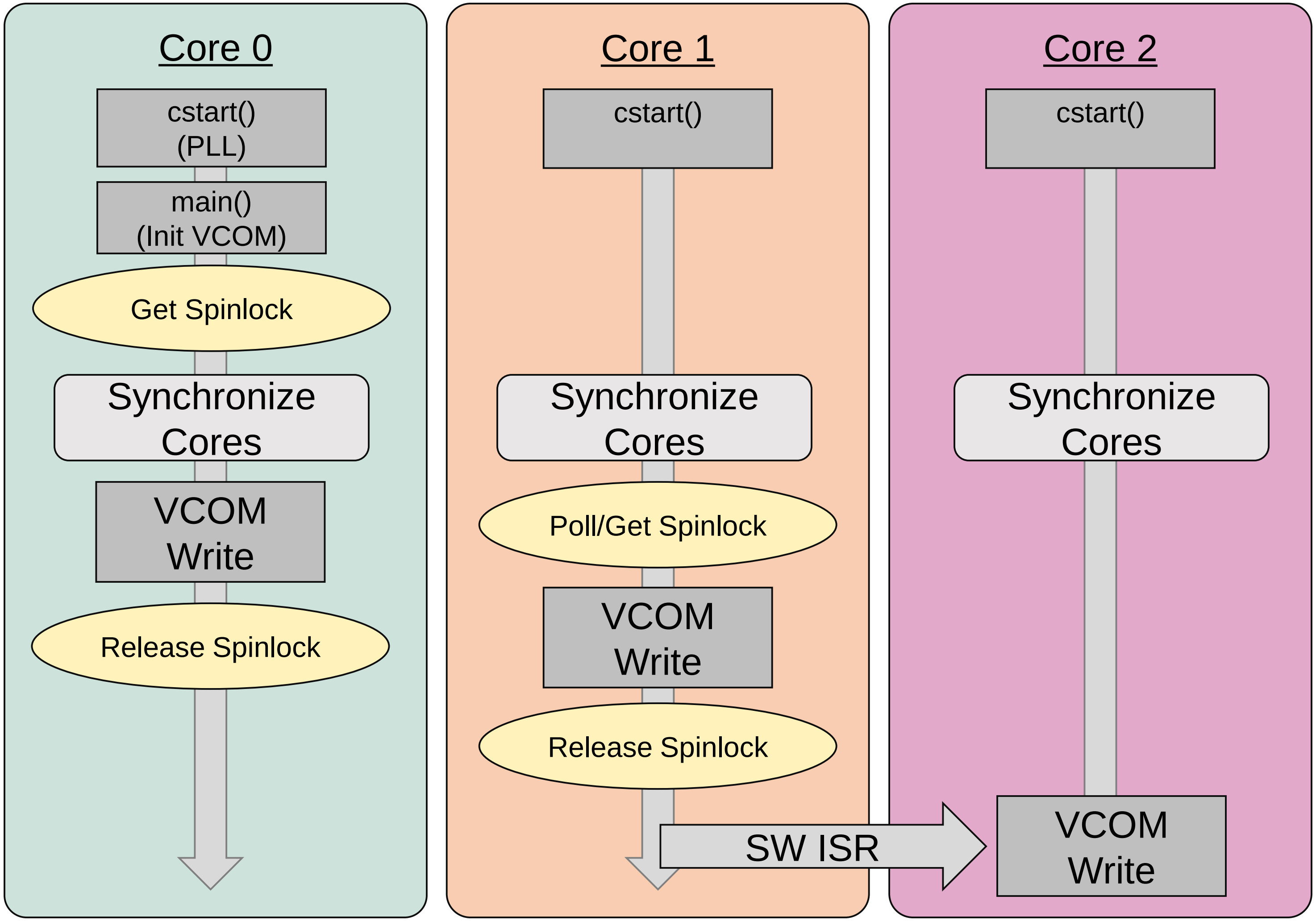
https://www.infineon.com/cms/en/product/microcontroller/32-bit-tricore-microcontroller

Under Free Tricore Tools subcategory:

Multicore_Example1.zip

TC277 Example

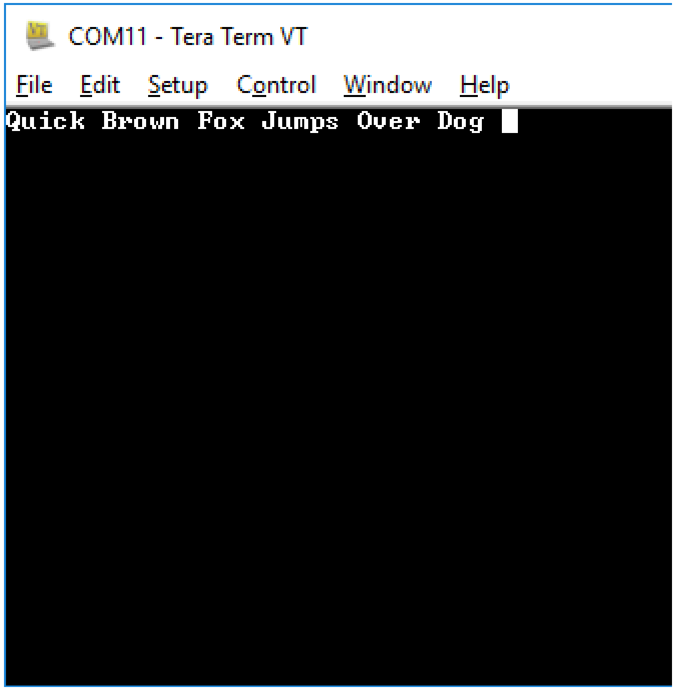
The example prints a sentence over UART/serial VCOM on the TC277 App Kit. Portions of the sentence are produced form each core. Synchronization is needed to produce the sentence in the right order.



TC277 Example

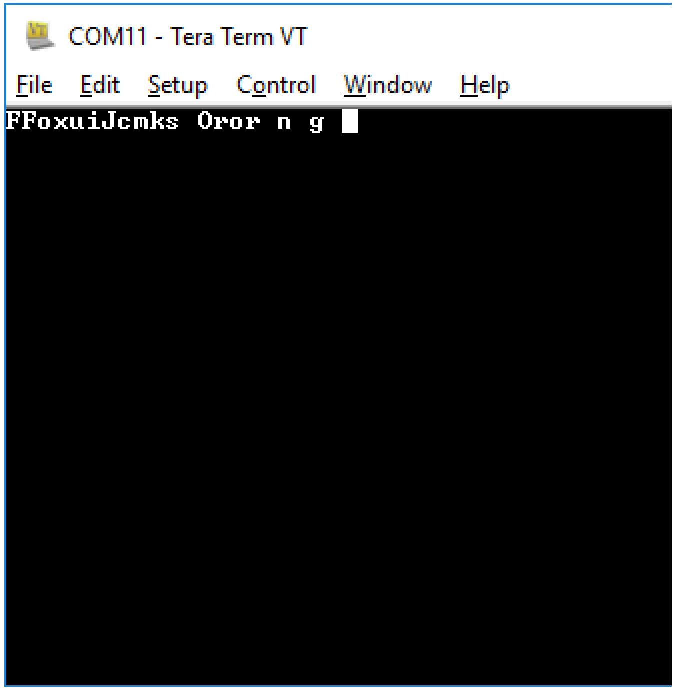
VCOM Baudrate 115200, 8 bit, no parity, 1 stop bit

USE_LOCKS == 1
(Correct Behavior)



```
Quick Brown Fox Jumps Over Dog
```

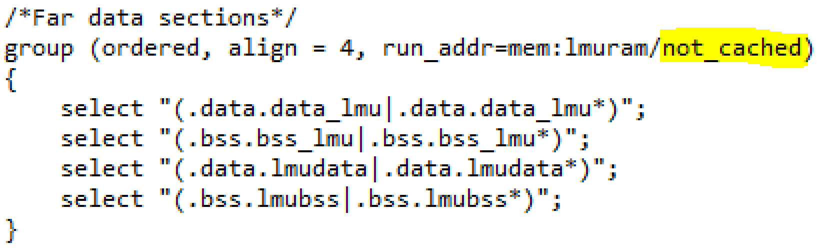
USE_LOCKS == 0
(Incorrect Behavior)



```
FFoxuiJcmks Oror n g
```

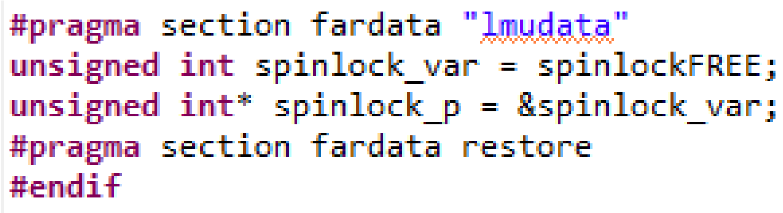
Location of Spinlock

The spinlock has been placed in the LMU, in the non-cached address space
- This allows for equal access for all the cores



```
/*Far data sections*/
group (ordered, align = 4, run_addr=mem:lmuram/not_cached)
{
    select "(.data.data_lmu|.data.data_lmu*)";
    select "(.bss.bss_lmu|.bss.bss_lmu*)";
    select "(.data.lmudata|.data.lmudata*)";
    select "(.bss.lmubss|.bss.lmubss*)";
}
```

And marked by pragma to force the memory location:

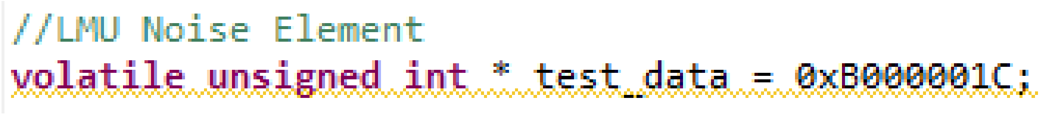


```
#pragma section fardata "lmudata"
unsigned int spinlock_var = spinlockFREE;
unsigned int* spinlock_p = &spinlock_var;
#pragma section fardata restore
#endif
```

Adding Noise to the LMU

We want to filter out the spinlock behavior in trace, but in order to prove the behavior we need to add some noise to the LMU observation block:

Noise Paramater Pointer Assignment



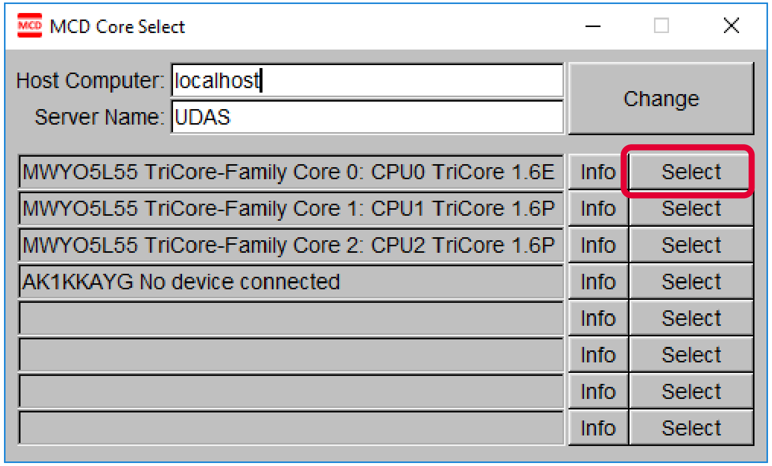
```
//LMU Noise Element
volatile unsigned int * test_data = 0xB000001C;
```

We will continuously update the memory location to “inject” noise in our trace

Using DAS MCD Basic Client

Open Windows Start -> DAS(64) -> MCD Basic Client

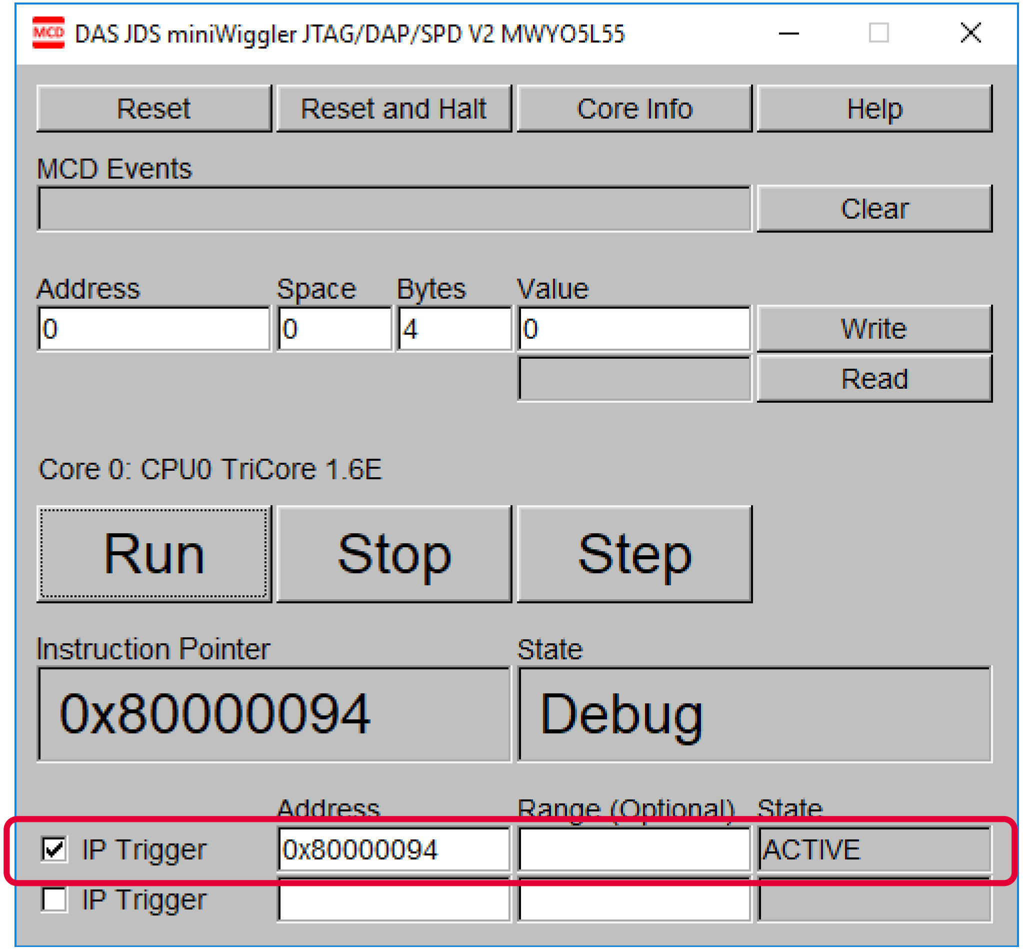
Let’s set a breakpoint in CPU0, at beginning of core0_main():



Host Computer: localhost

Server Name: UDAS

MWYO5L55 TriCore-Family Core 0: CPU0 TriCore 1.6E	Info	Select
MWYO5L55 TriCore-Family Core 1: CPU1 TriCore 1.6P	Info	Select
MWYO5L55 TriCore-Family Core 2: CPU2 TriCore 1.6P	Info	Select
AK1KKAYG No device connected	Info	Select
	Info	Select
	Info	Select
	Info	Select
	Info	Select



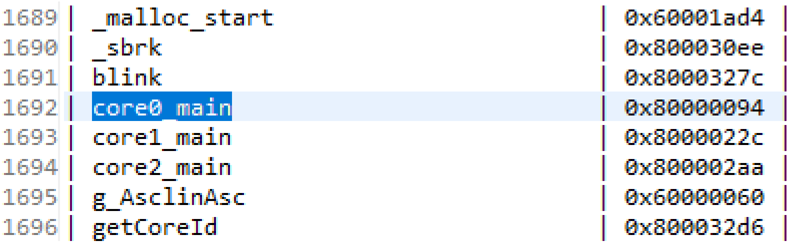
Address	Space	Bytes	Value
0	0	4	0

Core 0: CPU0 TriCore 1.6E

Instruction Pointer	State
0x80000094	Debug

	Address	Range (Optional)	State
☑ IP Trigger	0x80000094		ACTIVE
□ IP Trigger			

From Map File:



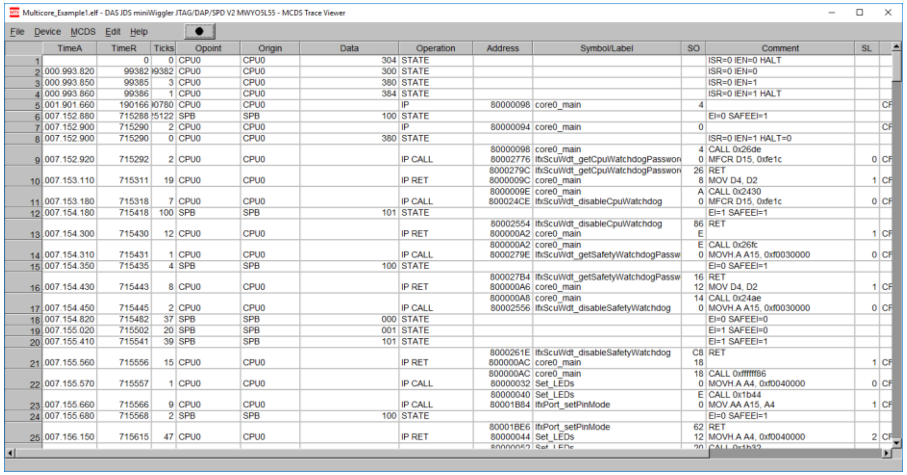
```
| _malloc_start                       | 0x60001ad4 |
| _sbrk                               | 0x800030ee |
| blink                               | 0x8000327c |
| core0_main                          | 0x80000094 |
| core1_main                          | 0x8000022c |
| core2_main                          | 0x800002aa |
| g_AsclinAsc                         | 0x60000060 |
| getCoreId                           | 0x800032d6 |
```

This is the starting point of when we want to begin the trace

Open MCDS Trace Viewer

In MCDS Menu

- File->Open .elf File..and browse for the elf (Debug folder of project)
- Device->Connect Device
- Hit the record button ● -> ●
- Hit Reset and Halt, then Run in MCD Basic Client, Trace Buffer is Filled:



Multicore_Example1.elf - DAS JDS miniWiggler JTAG/DAP/SPD V2 MWYO5L55 - MCDS Trace Viewer

	TimeA	TimeR	Ticks	Opoint	Origin	Data	Operation	Address	Symbol/Label	SO	Comment	SL	
1		0	0	CPU0	CPU0	304	STATE				ISR=0 IEN=0 HALT		
2	000.993.820	99382	9382	CPU0	CPU0	300	STATE				ISR=0 IEN=0		
3	000.993.850	99385	3	CPU0	CPU0	380	STATE				ISR=0 IEN=1		
4	000.993.860	99386	1	CPU0	CPU0	384	STATE				ISR=0 IEN=1 HALT		
5	001.901.660	190166	0780	CPU0	CPU0		IP	80000098	core0_main	4			CF
6	007.152.880	715288	5122	SPB	SPB	100	STATE				EI=0 SAFEEI=1		
7	007.152.900	715290	2	CPU0	CPU0		IP	80000094	core0_main	0			CF
8	007.152.900	715290	0	CPU0	CPU0	380	STATE				ISR=0 IEN=1 HALT=0		
9	007.152.920	715292	2	CPU0	CPU0		IP CALL	80000098 80002776	core0_main IfxScuWdt_getCpuWatchdogPasswor	4 0	CALL 0x26de MFCR D15, 0xfe1c	0	CF
10	007.153.110	715311	19	CPU0	CPU0		IP RET	8000279C 8000009C	IfxScuWdt_getCpuWatchdogPasswor core0_main	26 8	RET MOV D4, D2	1	CF
11	007.153.180	715318	7	CPU0	CPU0		IP CALL	8000009E 800024CE	core0_main IfxScuWdt_disableCpuWatchdog	A 0	CALL 0x2430 MFCR D15, 0xfe1c	0	CF
12	007.154.180	715418	100	SPB	SPB	101	STATE				EI=1 SAFEEI=1		
13	007.154.300	715430	12	CPU0	CPU0		IP RET	80002554 800000A2	IfxScuWdt_disableCpuWatchdog core0_main	86 E	RET	1	CF
14	007.154.310	715431	1	CPU0	CPU0		IP CALL	800000A2 8000279E	core0_main IfxScuWdt_getSafetyWatchdogPassw	E 0	CALL 0x26fc MOVH.A A15, 0xf0030000	0	CF
15	007.154.350	715435	4	SPB	SPB	100	STATE				EI=0 SAFEEI=1		
16	007.154.430	715443	8	CPU0	CPU0		IP RET	800027B4 800000A6	IfxScuWdt_getSafetyWatchdogPassw core0_main	16 12	RET MOV D4, D2	1	CF
17	007.154.450	715445	2	CPU0	CPU0		IP CALL	800000A8 80002556	core0_main IfxScuWdt_disableSafetyWatchdog	14 0	CALL 0x24ae MOVH.A A15, 0xf0030000	0	CF
18	007.154.820	715482	37	SPB	SPB	000	STATE				EI=0 SAFEEI=0		
19	007.155.020	715502	20	SPB	SPB	001	STATE				EI=1 SAFEEI=0		
20	007.155.410	715541	39	SPB	SPB	101	STATE				EI=1 SAFEEI=1		
21	007.155.560	715556	15	CPU0	CPU0		IP RET	8000261E 800000AC	IfxScuWdt_disableSafetyWatchdog core0_main	C8 18	RET	1	CF
22	007.155.570	715557	1	CPU0	CPU0		IP CALL	800000AC 80000032	core0_main Set_LEDs	18 0	CALL 0xffffff86 MOVH.A A4, 0xf0040000	0	CF
23	007.155.660	715566	9	CPU0	CPU0		IP CALL	80000040 80001B84	Set_LEDs IfxPort_setPinMode	E 0	CALL 0x1b44 MOV.AA A15, A4	1	CF
24	007.155.680	715568	2	SPB	SPB	100	STATE				EI=0 SAFEEI=1		
25	007.156.150	715615	47	CPU0	CPU0		IP RET	80001BE6 80000044	IfxPort_setPinMode Set_LEDs	62 12	RET MOVH.A A4, 0xf0040000	2	CF
								80000052	Set_LEDs	20	CALL 0x1b32		

Hint.. you can increase the trace buffer size in MCDS->General. By default, it is 16kB for the trace tile

Table 2-1 TC29/7/6/3xED Comparison

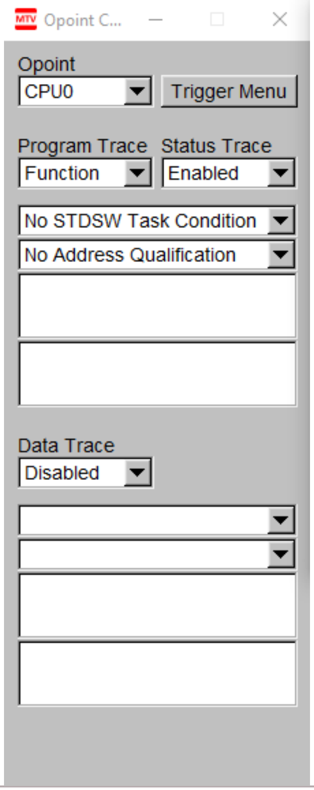
	9xED	7xED	6xED	3xED
PD features	3 CPUs	3 CPUs	2 CPUs	1 CPU
EMEM RAM total	2064 KB	1024 KB	528 KB	528 KB
EMEM TCM	1024 KB	1024 KB	512 KB	512 KB

Trace uses TCM →

Targeting the MCDS Trace

- By default, we are only targeting CPU0 as an observation point
- We are also targeting a “Flow” trace, entry and exit of functions and ISRs only (least amount of trace information)
- But we really want to look at data in this case. Let’s change the observation point (MCDS->Opoint CPUa and MCDS->Opoint SRI):

CPU0 setup remains
the same, as a reference.



Opoint: CPU0

Program Trace: Function

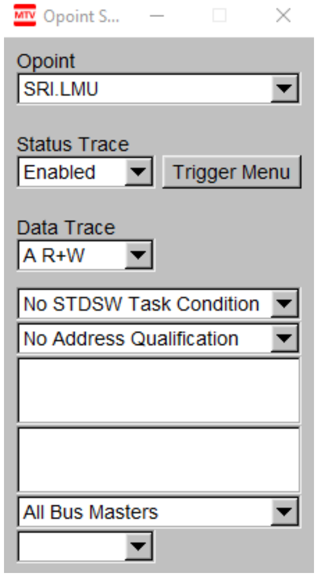
Status Trace: Enabled

No STDSW Task Condition

No Address Qualification

Data Trace: Disabled

Let’s observe the LMU
And all Reads and Writes.



Opoint: SRI.LMU

Status Trace: Enabled

Data Trace: A R+W

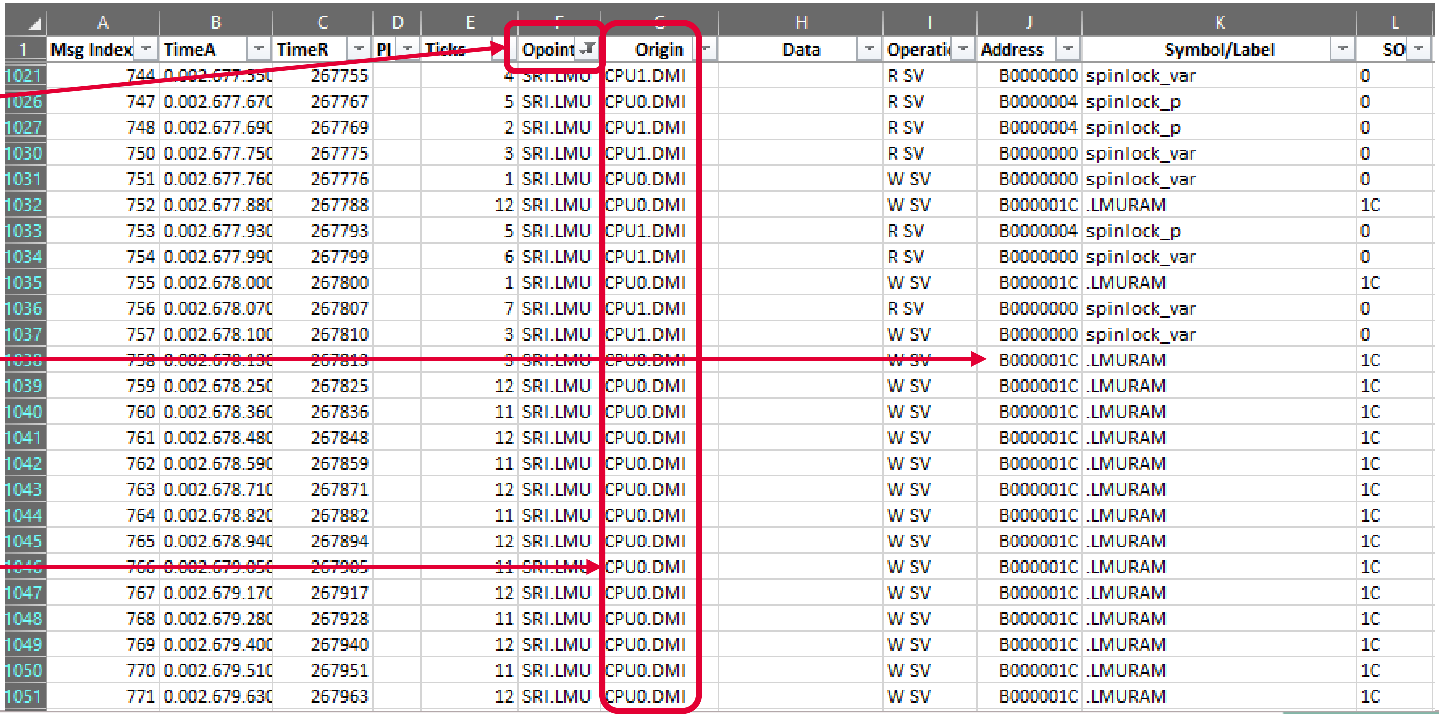
No STDSW Task Condition

No Address Qualification

All Bus Masters

Run Again with LMU observations Added

- In MCD Basic Client, Reset and Halt
- Run to core0_main
- Hit Record in MCDS Trace Viewer
- Run until Trace Buffer is Full
- In Trace Viewer, Edit->Select All, Edit->Copy, and paste into Excel Spreadsheet (Template provided in DAS tools, DAS(64)->MCDS Trace Viewer Template)



You can filter for just LMU activity

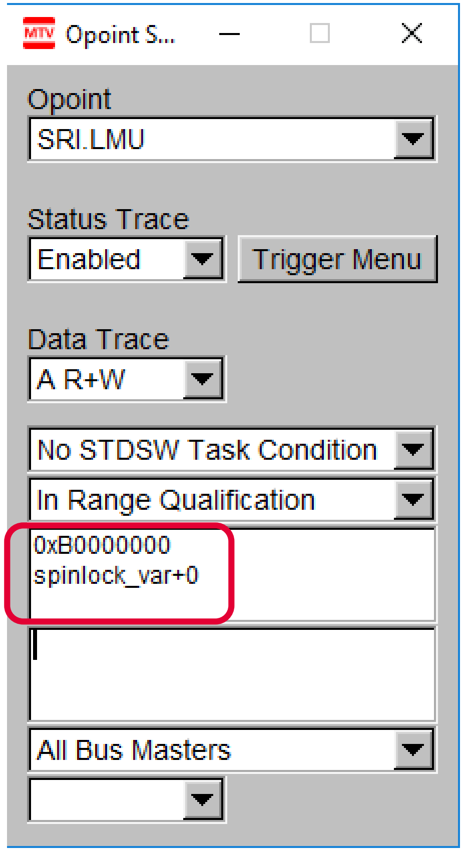
You can see our injected noise variable

CPU0 and CPU1 Activity

	A	B	C	D	E	F	G	H	I	J	K	L
1	Msg Index	TimeA	TimeR	PI	Ticks	Opoint	Origin	Data	Operati	Address	Symbol/Label	SO
1021	744	0.002.677.550	267755		4	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
1026	747	0.002.677.670	267767		5	SRI.LMU	CPU0.DMI		R SV	B0000004	spinlock_p	0
1027	748	0.002.677.690	267769		2	SRI.LMU	CPU1.DMI		R SV	B0000004	spinlock_p	0
1030	750	0.002.677.750	267775		3	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
1031	751	0.002.677.760	267776		1	SRI.LMU	CPU0.DMI		W SV	B0000000	spinlock_var	0
1032	752	0.002.677.880	267788		12	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1033	753	0.002.677.930	267793		5	SRI.LMU	CPU1.DMI		R SV	B0000004	spinlock_p	0
1034	754	0.002.677.990	267799		6	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
1035	755	0.002.678.000	267800		1	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1036	756	0.002.678.070	267807		7	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
1037	757	0.002.678.100	267810		3	SRI.LMU	CPU1.DMI		W SV	B0000000	spinlock_var	0
1038	758	0.002.678.130	267813		3	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1039	759	0.002.678.250	267825		12	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1040	760	0.002.678.360	267836		11	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1041	761	0.002.678.480	267848		12	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1042	762	0.002.678.590	267859		11	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1043	763	0.002.678.710	267871		12	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1044	764	0.002.678.820	267882		11	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1045	765	0.002.678.940	267894		12	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1046	766	0.002.679.050	267905		11	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1047	767	0.002.679.170	267917		12	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1048	768	0.002.679.280	267928		11	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1049	769	0.002.679.400	267940		12	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1050	770	0.002.679.510	267951		11	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C
1051	771	0.002.679.630	267963		12	SRI.LMU	CPU0.DMI		W SV	B000001C	.LMURAM	1C

Adding Range Qualification

- Now we will remove the noise variable by adding a range qualifier to the LMU observation point
- The spinlock is at 0xB0000000, in which now we can look at Reads and writes only at this address
- In this case we chose one variable, you can alternatively set a range, ie “0xB0000000 0xB0000008” will set a range of 2 32 bit words in LMU



Opoint: SRI.LMU

Status Trace: Enabled

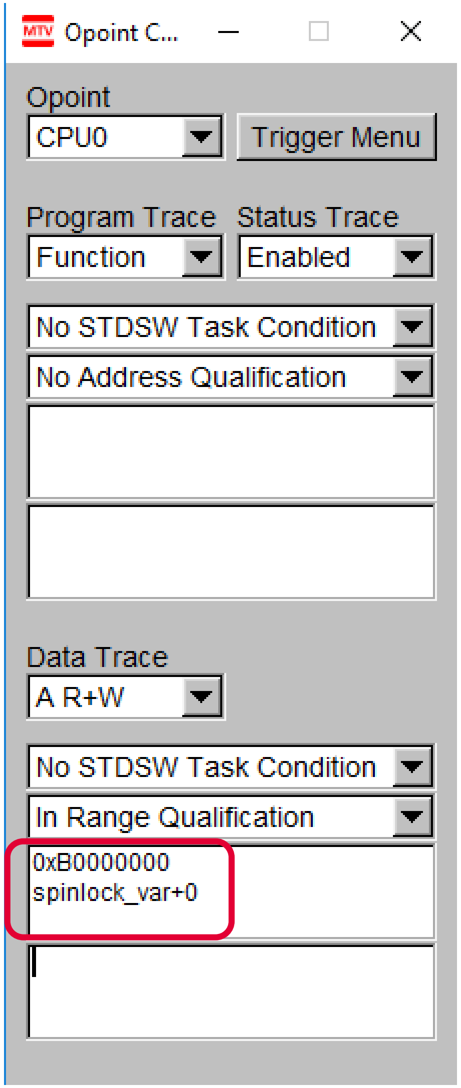
Data Trace: A R+W

No STDSW Task Condition

In Range Qualification

0xB0000000
spinlock_var+0

All Bus Masters



Opoint: CPU0

Program Trace: Function

Status Trace: Enabled

No STDSW Task Condition

No Address Qualification

Data Trace: A R+W

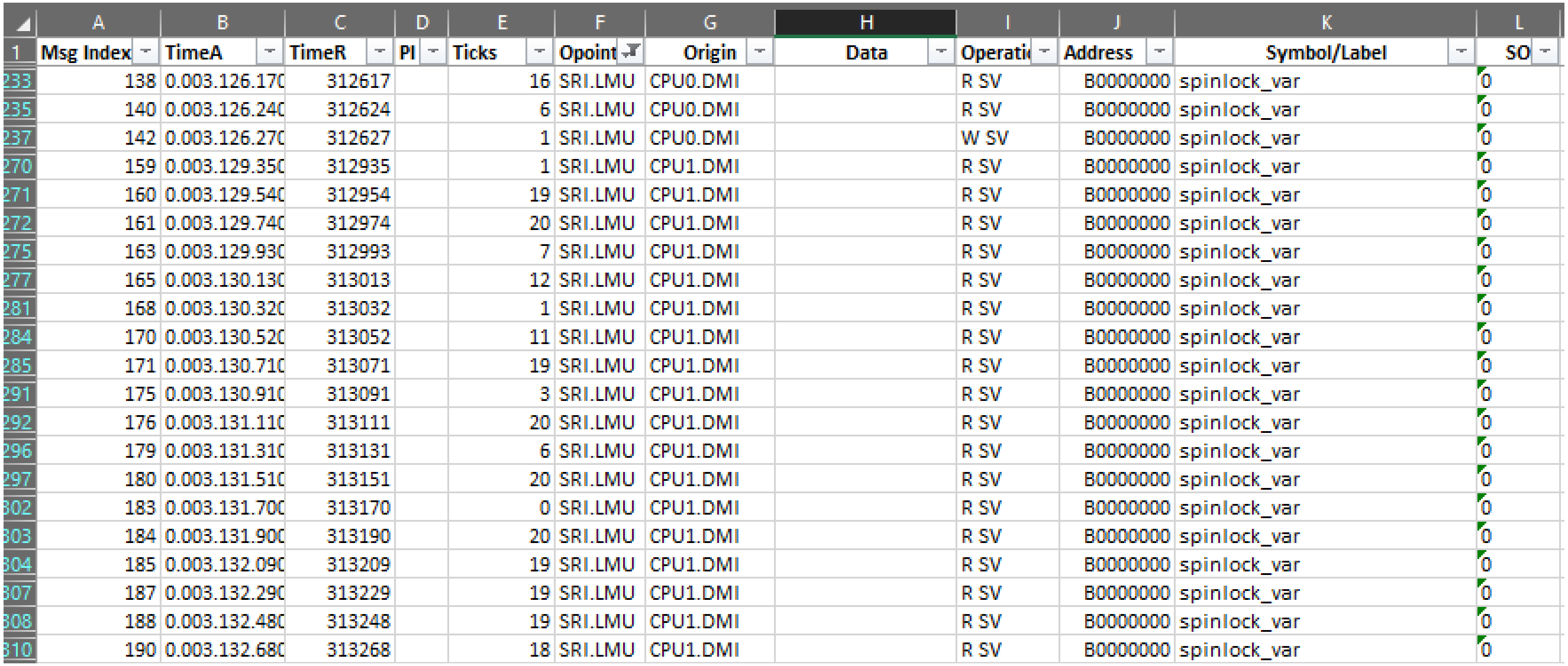
No STDSW Task Condition

In Range Qualification

0xB0000000
spinlock_var+0

Filtered Trace Buffer

› Noise Variable is Removed



	A	B	C	D	E	F	G	H	I	J	K	L
1	Msg Index	TimeA	TimeR	PI	Ticks	Opoint	Origin	Data	Operati	Address	Symbol/Label	SO
233	138	0.003.126.170	312617		16	SRI.LMU	CPU0.DMI		R SV	B0000000	spinlock_var	0
235	140	0.003.126.240	312624		6	SRI.LMU	CPU0.DMI		R SV	B0000000	spinlock_var	0
237	142	0.003.126.270	312627		1	SRI.LMU	CPU0.DMI		W SV	B0000000	spinlock_var	0
270	159	0.003.129.350	312935		1	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
271	160	0.003.129.540	312954		19	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
272	161	0.003.129.740	312974		20	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
275	163	0.003.129.930	312993		7	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
277	165	0.003.130.130	313013		12	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
281	168	0.003.130.320	313032		1	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
284	170	0.003.130.520	313052		11	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
285	171	0.003.130.710	313071		19	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
291	175	0.003.130.910	313091		3	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
292	176	0.003.131.110	313111		20	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
296	179	0.003.131.310	313131		6	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
297	180	0.003.131.510	313151		20	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
302	183	0.003.131.700	313170		0	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
303	184	0.003.131.900	313190		20	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
304	185	0.003.132.090	313209		19	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
307	187	0.003.132.290	313229		19	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
308	188	0.003.132.480	313248		19	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0
310	190	0.003.132.680	313268		18	SRI.LMU	CPU1.DMI		R SV	B0000000	spinlock_var	0

Targeting the MCDS Trace with Lauterbach (Data)

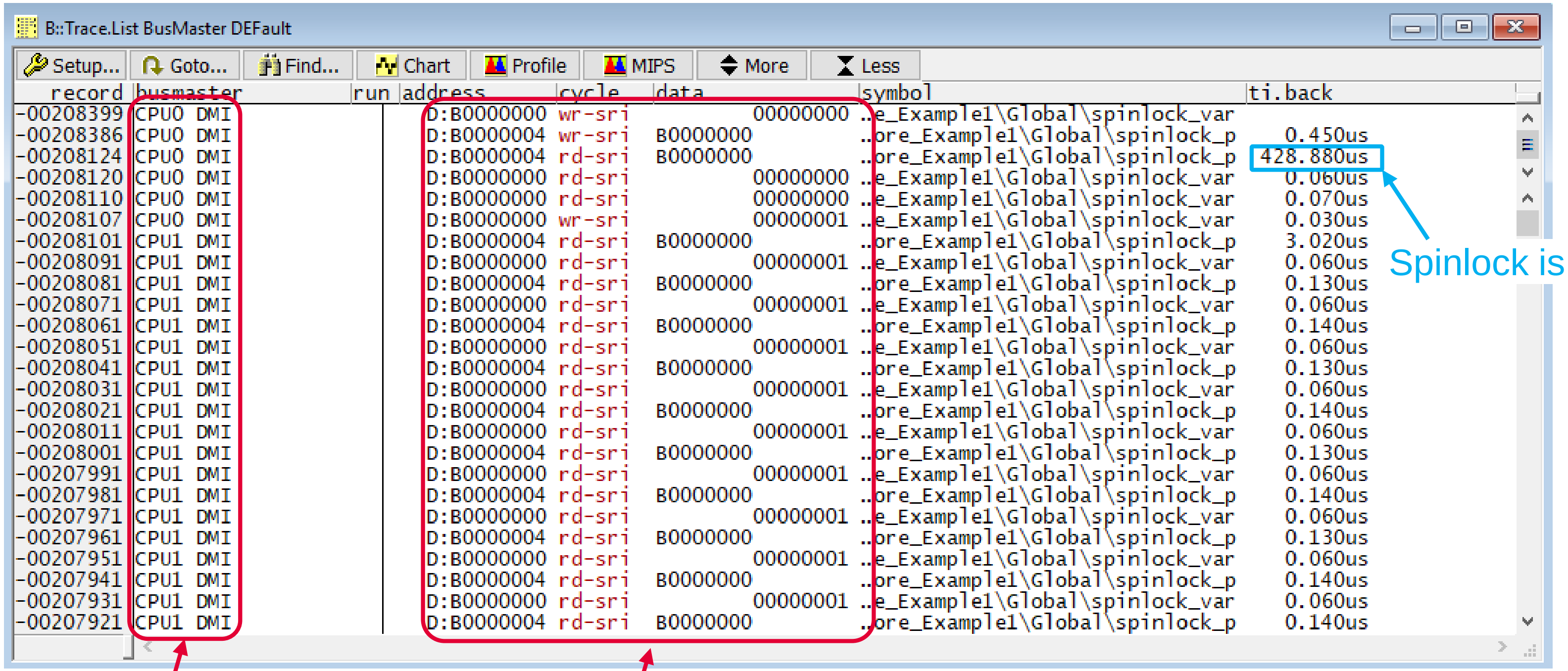
- We can start with the example script in Lauterbach folder (../demo/tricore/flash/tc…..cmm)
- Let’s target a Data Range:

Like before, let’s exclude the noise value at 0xB000001C

```
;Set a variable for a data range
&address_data = “0xB0000000—0xB0000008”
;Reset MCDS to default settings
MCDS.Reset
;Enable MCDS Timestamps
MCDS.Timestamp ON
;Enable Automatic Clock Calculation
CLOCK.ON
;Configure LMU as SRI 1 Observation point
MCDS.SOURCE.Set SRI.1.SLAVE LMU
;Configure SRI1 address pre-trigger (up to 8 available)
MCDS.Set SRI EAddr0_1 &address_data
;Configure pre-trigger to EVT (up to 16 available)
MCDS.Set SRI EVT0 EAddr0_1
; Set actions read-address, read-data, write-address and write-data to trigger when EVT0
MCDS.Set SRI ACT DTU_RADR1 aisAUTO EVT0
MCDS.Set SRI ACT DTU_RDAT1 aisAUTO EVT0
MCDS.Set SRI ACT DTU_WADR1 aisAUTO EVT0
MCDS.Set SRI ACT DTU_WDAT1 aisAUTO EVT0
```

A Look at the Trace Data

› Based on previous settings, we now can look at a specific range in the LMU, every time it is accessed.



```
   record busmaster       run address    cycle   data              symbol                                ti.back
-00208399 CPU0  DMI           D:B0000000 wr-sri           00000000 ..e_Example1\Global\spinlock_var
-00208386 CPU0  DMI           D:B0000004 wr-sri   B0000000         ..ore_Example1\Global\spinlock_p     0.450us
-00208124 CPU0  DMI           D:B0000004 rd-sri   B0000000         ..ore_Example1\Global\spinlock_p   428.880us
-00208120 CPU0  DMI           D:B0000000 rd-sri           00000000 ..e_Example1\Global\spinlock_var     0.060us
-00208110 CPU0  DMI           D:B0000000 rd-sri           00000000 ..e_Example1\Global\spinlock_var     0.070us
-00208107 CPU0  DMI           D:B0000000 wr-sri           00000001 ..e_Example1\Global\spinlock_var     0.030us
-00208101 CPU1  DMI           D:B0000004 rd-sri   B0000000         ..ore_Example1\Global\spinlock_p     3.020us
-00208091 CPU1  DMI           D:B0000000 rd-sri           00000001 ..e_Example1\Global\spinlock_var     0.060us
-00208081 CPU1  DMI           D:B0000004 rd-sri   B0000000         ..ore_Example1\Global\spinlock_p     0.130us
-00208071 CPU1  DMI           D:B0000000 rd-sri           00000001 ..e_Example1\Global\spinlock_var     0.060us
-00208061 CPU1  DMI           D:B0000004 rd-sri   B0000000         ..ore_Example1\Global\spinlock_p     0.140us
-00208051 CPU1  DMI           D:B0000000 rd-sri           00000001 ..e_Example1\Global\spinlock_var     0.060us
-00208041 CPU1  DMI           D:B0000004 rd-sri   B0000000         ..ore_Example1\Global\spinlock_p     0.130us
-00208031 CPU1  DMI           D:B0000000 rd-sri           00000001 ..e_Example1\Global\spinlock_var     0.060us
-00208021 CPU1  DMI           D:B0000004 rd-sri   B0000000         ..ore_Example1\Global\spinlock_p     0.140us
-00208011 CPU1  DMI           D:B0000000 rd-sri           00000001 ..e_Example1\Global\spinlock_var     0.060us
-00208001 CPU1  DMI           D:B0000004 rd-sri   B0000000         ..ore_Example1\Global\spinlock_p     0.130us
-00207991 CPU1  DMI           D:B0000000 rd-sri           00000001 ..e_Example1\Global\spinlock_var     0.060us
-00207981 CPU1  DMI           D:B0000004 rd-sri   B0000000         ..ore_Example1\Global\spinlock_p     0.140us
-00207971 CPU1  DMI           D:B0000000 rd-sri           00000001 ..e_Example1\Global\spinlock_var     0.060us
-00207961 CPU1  DMI           D:B0000004 rd-sri   B0000000         ..ore_Example1\Global\spinlock_p     0.130us
-00207951 CPU1  DMI           D:B0000000 rd-sri           00000001 ..e_Example1\Global\spinlock_var     0.060us
-00207941 CPU1  DMI           D:B0000004 rd-sri   B0000000         ..ore_Example1\Global\spinlock_p     0.140us
-00207931 CPU1  DMI           D:B0000000 rd-sri           00000001 ..e_Example1\Global\spinlock_var     0.060us
-00207921 CPU1  DMI           D:B0000004 rd-sri   B0000000         ..ore_Example1\Global\spinlock_p     0.140us
```

Spinlock is set here!

Who wrote to it:
Bus Master and Interface
(ie DMI, Data Memory Interface)

Address, Action, Data

Symbol

Time

Targeting the MCDS Trace with Lauterbach (Program)

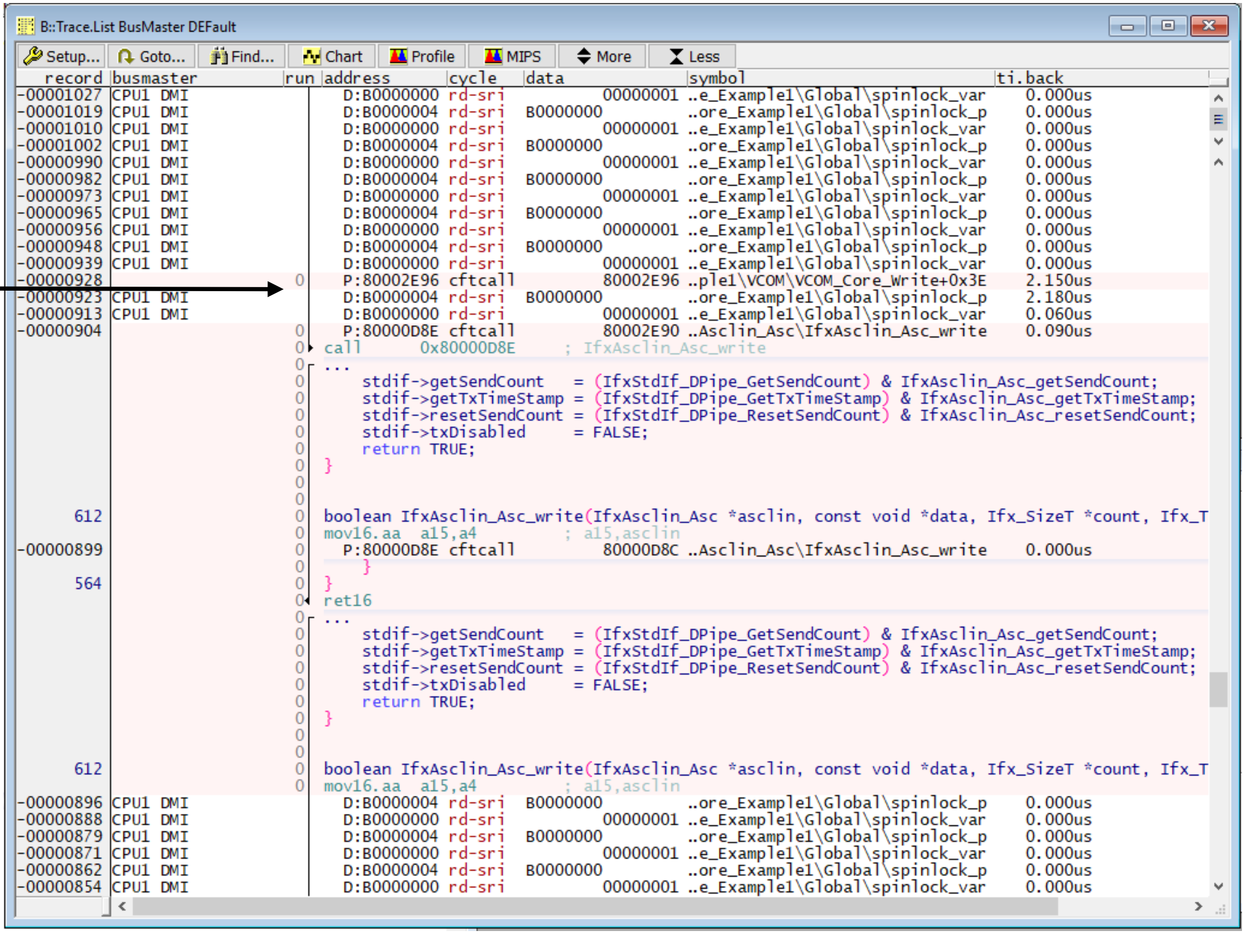
- Let’s target a Program Range:

Let’s pick the VCOM_Core_Write as my target function

```
;Set address range from map file
&address_program = “0x80002E58—0xB0002F58”
;Reset MCDS to default settings
MCDS.Reset
;Enable MCDS Timestamps
MCDS.Timestamp ON
;Enable Automatic Clock Calculation
CLOCK.ON
;Configure CPU0 Observation point
MCDS.SOURCE.Set CpuMux0.Core Tricore0
;Set Program Trace ON
MCDS.SOURCE.CpuMux0.Program ON
;Set Program Trace Mode to CFT Compact Function Trace
MCDS.SOURCE.CpuMux0.PTMODE CFT
;Enable the Trace for the given range
Break.Set &address_program /Program /TraceEnable
;Include in the Trace List
Trace.List BusMaster DEFault
```

A Look at the Trace Data

VCOM_Core_Write Address Range Triggers CFT into Trace Buffer



```
   record busmaster         run address      cycle    data       symbol                                 ti.back
-00001027 CPU1  DMI               D:B0000000 rd-sri            00000001 ..e_Example1\Global\spinlock_var      0.000us
-00001019 CPU1  DMI               D:B0000004 rd-sri   B0000000          ..ore_Example1\Global\spinlock_p      0.000us
-00001010 CPU1  DMI               D:B0000000 rd-sri            00000001 ..e_Example1\Global\spinlock_var      0.000us
-00001002 CPU1  DMI               D:B0000004 rd-sri   B0000000          ..ore_Example1\Global\spinlock_p      0.000us
-00000990 CPU1  DMI               D:B0000000 rd-sri            00000001 ..e_Example1\Global\spinlock_var      0.000us
-00000982 CPU1  DMI               D:B0000004 rd-sri   B0000000          ..ore_Example1\Global\spinlock_p      0.000us
-00000973 CPU1  DMI               D:B0000000 rd-sri            00000001 ..e_Example1\Global\spinlock_var      0.000us
-00000965 CPU1  DMI               D:B0000004 rd-sri   B0000000          ..ore_Example1\Global\spinlock_p      0.000us
-00000956 CPU1  DMI               D:B0000000 rd-sri            00000001 ..e_Example1\Global\spinlock_var      0.000us
-00000948 CPU1  DMI               D:B0000004 rd-sri   B0000000          ..ore_Example1\Global\spinlock_p      0.000us
-00000939 CPU1  DMI               D:B0000000 rd-sri            00000001 ..e_Example1\Global\spinlock_var      0.000us
-00000928                     0   P:80002E96 cftcall           80002E96 ..ple1\VCOM\VCOM_Core_Write+0x3E      2.150us
-00000923 CPU1  DMI               D:B0000004 rd-sri   B0000000          ..ore_Example1\Global\spinlock_p      2.180us
-00000913 CPU1  DMI               D:B0000000 rd-sri            00000001 ..e_Example1\Global\spinlock_var      0.060us
-00000904                     0   P:80000D8E cftcall           80002E90 ..Asclin_Asc\IfxAsclin_Asc_write      0.090us
                              0 call       0x80000D8E     ; IfxAsclin_Asc_write
                              0 ...
                              0     stdif->getSendCount   = (IfxStdIf_DPipe_GetSendCount) & IfxAsclin_Asc_getSendCount;
                              0     stdif->getTxTimeStamp = (IfxStdIf_DPipe_GetTxTimeStamp) & IfxAsclin_Asc_getTxTimeStamp;
                              0     stdif->resetSendCount = (IfxStdIf_DPipe_ResetSendCount) & IfxAsclin_Asc_resetSendCount;
                              0     stdif->txDisabled     = FALSE;
                              0     return TRUE;
                              0 }
                              0
                              0
      612                     0 boolean IfxAsclin_Asc_write(IfxAsclin_Asc *asclin, const void *data, Ifx_SizeT *count, Ifx_T
                              0 mov16.aa  a15,a4          ; a15,asclin
-00000899                     0   P:80000D8E cftcall           80000D8C ..Asclin_Asc\IfxAsclin_Asc_write      0.000us
                              0     }
      564                     0 }
                              0 ret16
                              0 ...
                              0     stdif->getSendCount   = (IfxStdIf_DPipe_GetSendCount) & IfxAsclin_Asc_getSendCount;
                              0     stdif->getTxTimeStamp = (IfxStdIf_DPipe_GetTxTimeStamp) & IfxAsclin_Asc_getTxTimeStamp;
                              0     stdif->resetSendCount = (IfxStdIf_DPipe_ResetSendCount) & IfxAsclin_Asc_resetSendCount;
                              0     stdif->txDisabled     = FALSE;
                              0     return TRUE;
                              0 }
                              0
                              0
      612                     0 boolean IfxAsclin_Asc_write(IfxAsclin_Asc *asclin, const void *data, Ifx_SizeT *count, Ifx_T
                              0 mov16.aa  a15,a4          ; a15,asclin
-00000896 CPU1  DMI               D:B0000004 rd-sri   B0000000          ..ore_Example1\Global\spinlock_p      0.000us
-00000888 CPU1  DMI               D:B0000000 rd-sri            00000001 ..e_Example1\Global\spinlock_var      0.000us
-00000879 CPU1  DMI               D:B0000004 rd-sri   B0000000          ..ore_Example1\Global\spinlock_p      0.000us
-00000871 CPU1  DMI               D:B0000000 rd-sri            00000001 ..e_Example1\Global\spinlock_var      0.000us
-00000862 CPU1  DMI               D:B0000004 rd-sri   B0000000          ..ore_Example1\Global\spinlock_p      0.000us
-00000854 CPU1  DMI               D:B0000000 rd-sri            00000001 ..e_Example1\Global\spinlock_var      0.000us
```

infineon

Part of your life. Part of tomorrow.