### Real Insight from Code to Silicon

# SourcePoint ScanWorks

### WinDbg JTAG-based debug with DCI and EXDI

	2				
2	а.	SourcePoint v7.12.0 IDCI1 - T	igerLake -	:\Users\alans\Documents\Arium\SourcePoint-IA 7.12.33\IGL03.	DI
					and the second sec

File Edit View Processor Options Window Help



썗 Load UEFI Macros 🖏 썗 NPK 뺿 EnableTH 🖏 shell command 🖓 include command 👘 🔟 迫

🖶 R 🖪 O. O. 🗋 🔹 📲



						- 0 ×
Breakpoints (	🕀 Code 💙 Comman	nd 🔝 Lo	g 🛄 Mem	ory IP Registers 🔍 Sym	bols 🦻 Trace	👀 Viewpoint 🔍 Watch
						9
		00 Viewp	oint			
		Nan	ne	Description		Status
		P0	Tige	rLake	Stopped	
		O P1	Tige	rLake	Stopped	
		O P2	Tige	rLake	Stopped	
		O P3	Tige	rLake	Stopped	
		I				
				General Registers (P0*)		
					Name	Value
				⊟-Intel 64	RAX	FFFFF8027EEE56E0
				General	RBX	000000000000000000000000000000000000000
				-Floating Point	RUA	FFFFC4015C3F8210
				Segment	RDA	FFFFC4015C61F010
				Centrel	RDF	FFFFF9025643319
	Addrose		Value		RDT	000000000000000000000000000000000000000
	Address		value	Debug	RSP	FFFFF8025E944EA
	PPPPPPAAAFA(1A)	TOT	-		R8	000000000000000000000000000000000000000
	FFFFF802576A8.	SEUL		YMM - SP	R9	FFFFC4015C3F81B
	FFFFF802576A83	348L		YMM - DP	R10	FFFFF8027EEE15A
nmon	FFFFF802576A83	36CL		YMM - Int	R11	000000000000000000000000000000000000000
	FFFFF802576A78	388L		. MSR	R12	00000000FFFFFFF
	FFFFF802576A87	740L		User	R13	000000048F82804
	FFFFF802576A86	SCOL			R14	000000000000000000000000000000000000000
	FFFFF802576A84	ACOL		1	RIS	FFFFC4015Cb1E000
renmoveh	FFFFF802576480	TOOT	1	-	DG	0010
Tepinovau	FFFFF902576A00	TOOT		-	55	0018
	FFFFF002576800	COL	-	-	FS	002B
277	FFFFF802576A84	ACOT		-	FS	0053
ie	FFFFF802576A84	490L	-	-	GS	002B
	FFFFF802576C90	008L		-	RIP	FFFFF8027EEE4003
	FFFFF802576A00	028L			RFLAGS	000000000005024
	FFFFF802576738	324L				
eneric	FFFFF8025767F7	740L				
sunc i210	FFFFF8025768C	760L				
SYNC 12 10	FFFFF80257679E	TOOL				
ric	EFFERONAL 24 201	784L				
ric addr ge	rrrrr0025/6/91	WTA AND THE REAL PROPERTY OF				
ric _addr_ge ift_generic	FFFFF80257679	054L				
ric addr_ge ift_generic	FFFFF8025767C0	054L		~		

ScanWorks<sup>®</sup>

### Agenda

- WinDbg
- SourcePoint
- Why combine the two?
  - Enhancing WinDbg with JTAG-based run-control and trace features
  - "Debugging the Undebuggable"
- Demo configuration
  - What you'll see in the demo
- Demo
- Wrap-Up







# WinDbg

- WinDbg is a kernel-mode and user-mode debugger that's included in Debugging Tools for Windows
- De facto standard for Windows system-level programming/ debugging
- Very powerful: OS-aware (processes, threads, jobs, kernel symbols, etc.)









### SourcePoint

- Best-in-class UEFI debugger
- Support for x86: Intel (all CPUs) and AMD (EPYC)
- Source-level symbolic debugger, full run-control (stop, go, singlestep, breakpoints, etc.)
- Supports Advanced (Conditional)
  Breakpoints on AMD
- Supports innovative Trace features on Intel





ScanWorks<sup>®</sup>

## Why combine WinDbg and SourcePoint?

- Microsoft released update to EXDI (Extended Debug Interface)
- EXDI is an adaptation layer between a software debugger and a debugging target.
- Extends WinDbg by adding support for hardware-based debuggers (i.e. JTAG-based)
- WinDbg is the controller; SourcePoint is the worker

"Debugging the Undebuggable" https://www.andreaallievi.com/blog/debugging-theundebuggable-part-1/ **But on steroids!** 







### Why combine SourcePoint with WinDbg?

# Take advantage of powerful features of both applications:

### WinDbg

- OS-aware
- Extensible
- Go-to tool for kernel debugging

### SourcePoint

- Advanced breakpoint support (SMM entry/exit/data access, machine check, etc.)
- Intel Processor Trace
- Trace Hub
- Architectural Event Trace (AET)





### Intel Processor Trace

- **Execution Trace**
- Real-time (nominal performance impact)
- Stored in system memory (post-MRC)
- Call Tree, Call Chart, all with Search capability
- Up to 2GB instruction trace







### Intel Trace Hub

- Logic that comprises trace sources, a global hub with timestamp, trace destinations, and a trigger unit
- A slave device for writes from cores and any other trace sources
- Acts as a PCI device, and aligned with industry standards
- Sources include Architectural Event Trace (AET), ME Trace, SW/FW Trace, etc.
- Trace destinations include:
  - MTB (8kB, out of reset)
  - System Memory (after MRC)
  - Direct Connect Interface (out of reset, supports streaming trace)



Platform for Software Debug and Trace





# Architectural Event Trace (AET)

### Event Trace, that complements Instruction Trace. Requires JTAG.

Event Type	Event SubTypes
HW/SW Interrupt	HW_INTR
IRET	IRET
Exception	Exception
MSR	RDMSR, WRMSR
ΙΟ	PORT_IN, PORT_OUT, PORT_IN_ADDR
SGX	AEX, EENTER, ERESUME, EEXIT
CODE_BP	CODE_BP
DATA_BP	DATA_BP
FIXED_INT	SMI, RSM, NMI
SW_POWER	MONITOR/MWAIT
WBINVD	WBINVD_BEGIN, WBINVD_END



### Description

- HW interrupt trace
- **IRET trace**
- Exception, fault, trap trace
- MSR trace
- IO trace
- SGX trace
- Code breakpoint trace
- Data breakpoint trace
- "Fixed" interrupt trace
- MONITOR/MWAIT trace
- Write-back invalidate trace



## AET Tips #1

- **Probe-mode (JTAG) needed to initialize AET use outside** of probe mode (i.e. BIOS, device driver) causes #GP.
- AET is implemented in CPU microcode and does not modify the architectural behavior of the processors – no need to instrument code!
  - \* Enabling CODE/DATA\_BP changes the behavior of normal breakpoints – causes a trace event rather than a debug exception. Great for critical sections of code, concurrency issues, debugging memory accesses, etc.
- This is event trace, not instruction trace: source code/ symbols not required (but it's great if you have them!)







## AET Tips #2

- A Last Branch Record (LBR) instruction trace stack can be added to all event traces – a fast way to trace back ~ 300 instructions
  - LBR uses MSRs to track from\_address and to\_address pairs, so operates out of reset – no need for system memory Intel Processor Trace and AET can run concurrently
- Intel PT places trace data in system memory
- On Ice Lake processors and later, both AET LBR tracing and Intel Processor Trace can be enabled at the same time





### The Demo – what you'll see

- 1. SourcePoint makes JTAG connection at reset vector with DCI
- COTS Tiger Lake target booted from 2. reset vector to Windows desktop
- SourcePoint launches WinDbg, 3. makes EXDI connection
- Symbols visible in both WinDbg & 4. SourcePoint
- 5. Demonstrate run-control, symbolic debug (stop, go, set breakpoint, single-step, etc.), Intel Processor Trace, AET, etc.

### SourcePoint WinDbg







### DCI (USB) Cable



### **AAEON UP Xtreme i11 Tiger Lake**







ScanWorks® Platform for Embedded Instruments

### Resources

- SourcePoint Academy: <u>https://www.asset-</u> intertech.com/resources/academy/sourcepoint -academy/
  - SourcePoint WinDbg Getting Started Guide
  - Getting Started Guide for the AAEON UP Xtreme i11
  - Videos, Online Help, Release Notes, etc.





### **Questions and Contact Information**



### alan.sguigna@asset-intertech.com X DM @AlanSguigna



16

© 2023, ASSET InterTech, Inc.



### Real Insight from Code to Silicon

ASSET



© 2023, ASSET InterTech, Inc.,

