

NEXUS 5001 Forum Debug Interface Standard

www.ieee-isto.org/Nexus5001/



Gerhard Martin

Founding Member of NEXUS Technical Committee

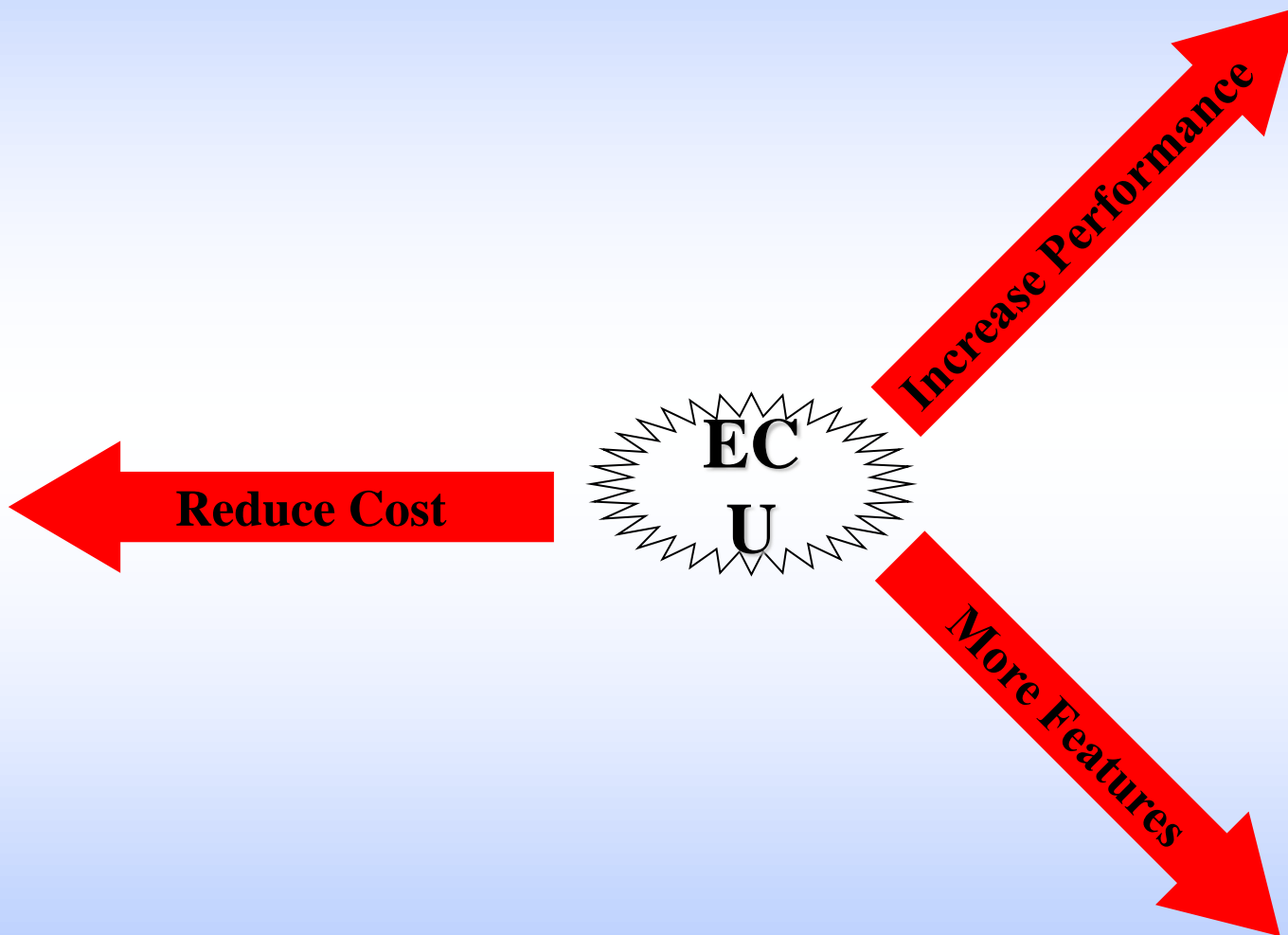
Infineon Technologies AG

Presentation Topics

- **Introduction**
- **Development Features**
- **Hardware Interface**
- **Software Interface**
- **Board/Connector Interface**
- **Summery**
- **Question/Answers**

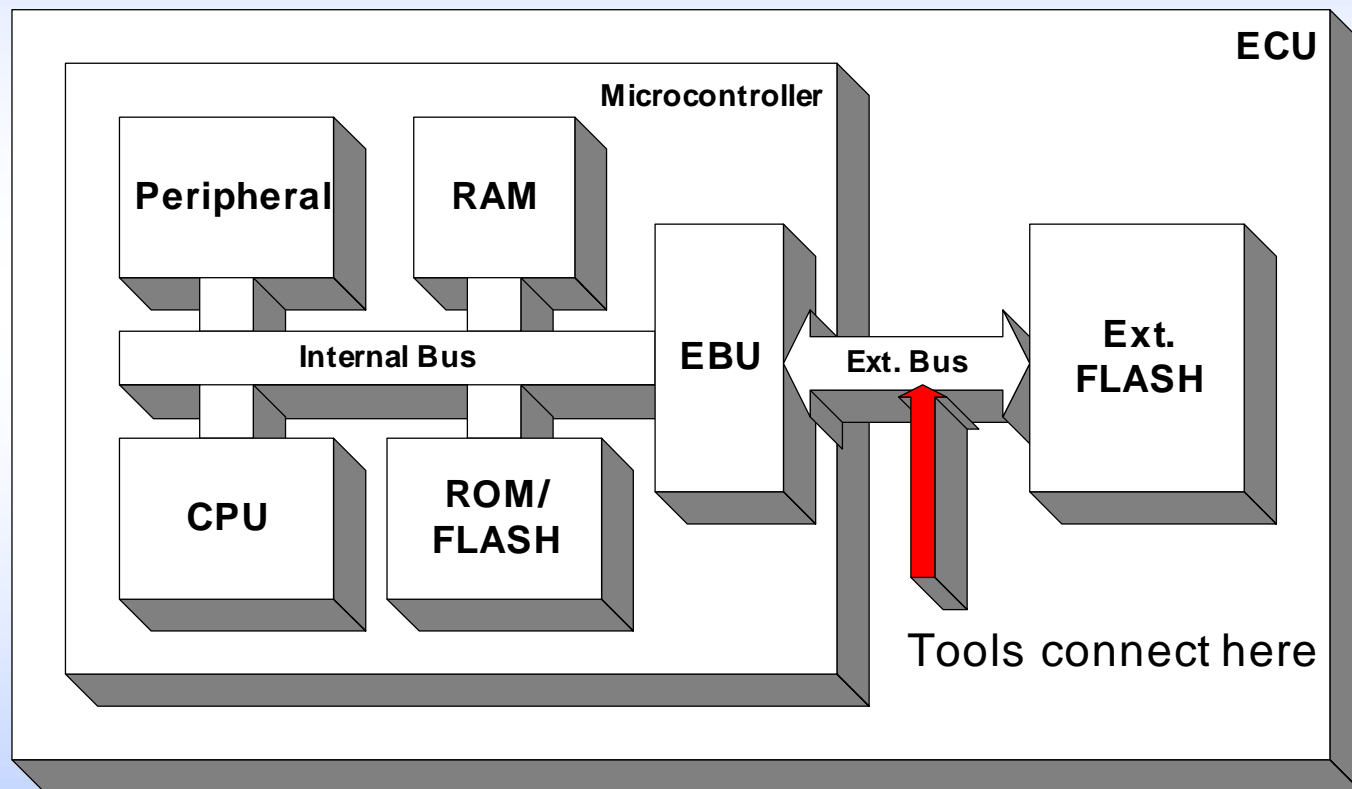
Introduction, Why NEXUS is needed

- **Introduction**
- **NEXUS Consortium**
- **Development Features**
- **Hardware Interface**
- **Software Interface**
- **Board/Connector Interface**
- **Summery**
- **Question/Answers**



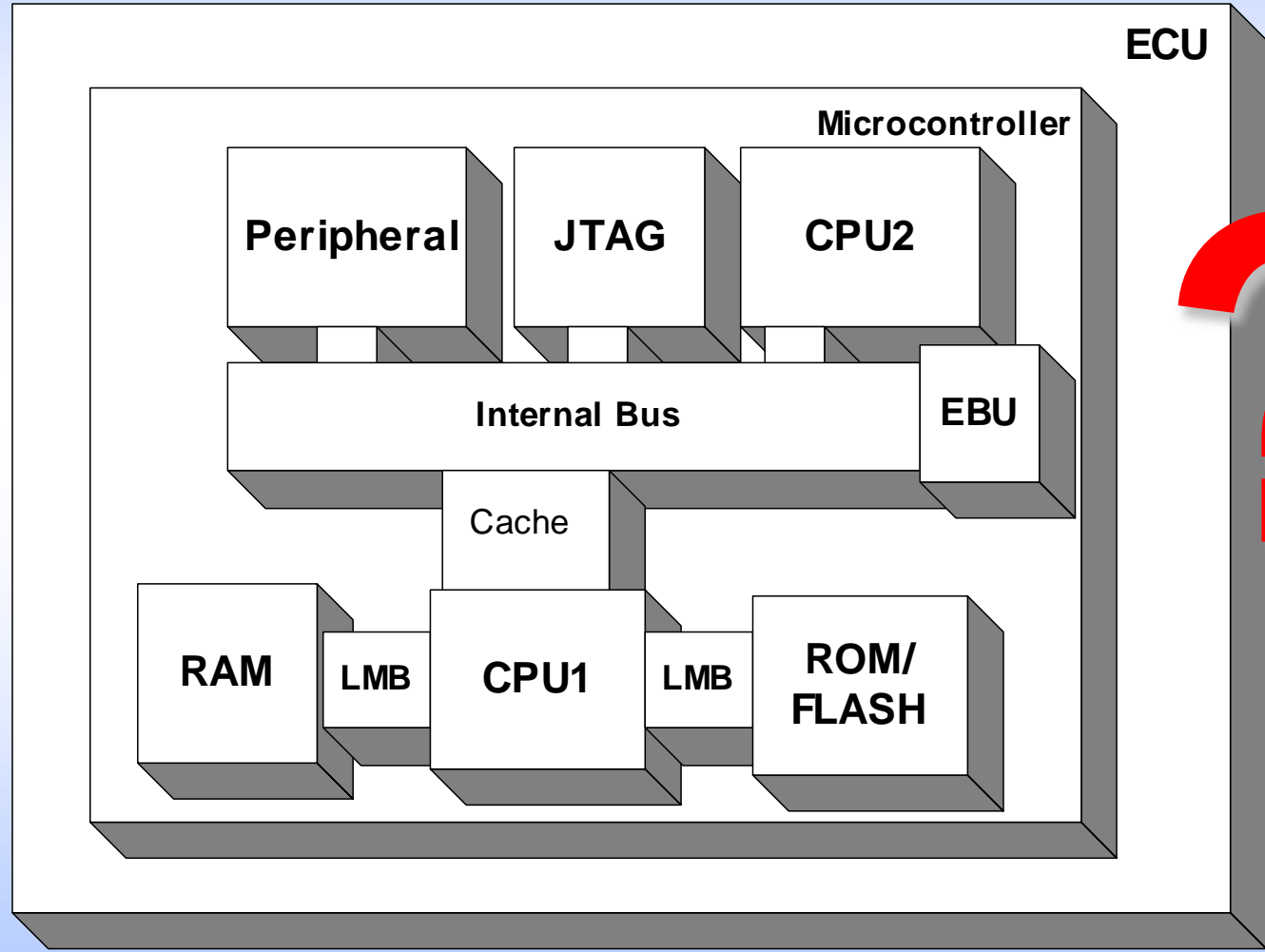
Introduction, Conventional ECU Design

- **Introduction**
- NEXUS Consortium
- Development Features
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/ Answers



Introduction, Tomorrows ECU's

- **Introduction**
- NEXUS Consortium
- Development Features
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summery
- Question/ Answers



- Introduction
- **NEXUS Consortium**
- Development Features
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers

The NEXUS Consortium

Current IEEE-ISTO NEXUS 5001 Membership List

- Introduction
- **NEXUS Consortium**
- Development Features
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers

Silicon Vendors:

- ☑ [Hitachi Semiconductor Inc.](#)
- ☑ [Infineon Technologies AG](#)
- ☑ Mitsubishi Electric Corp.
- ☑ [Motorola](#)
- ☑ ST Microelectronics

Tool Vendors:

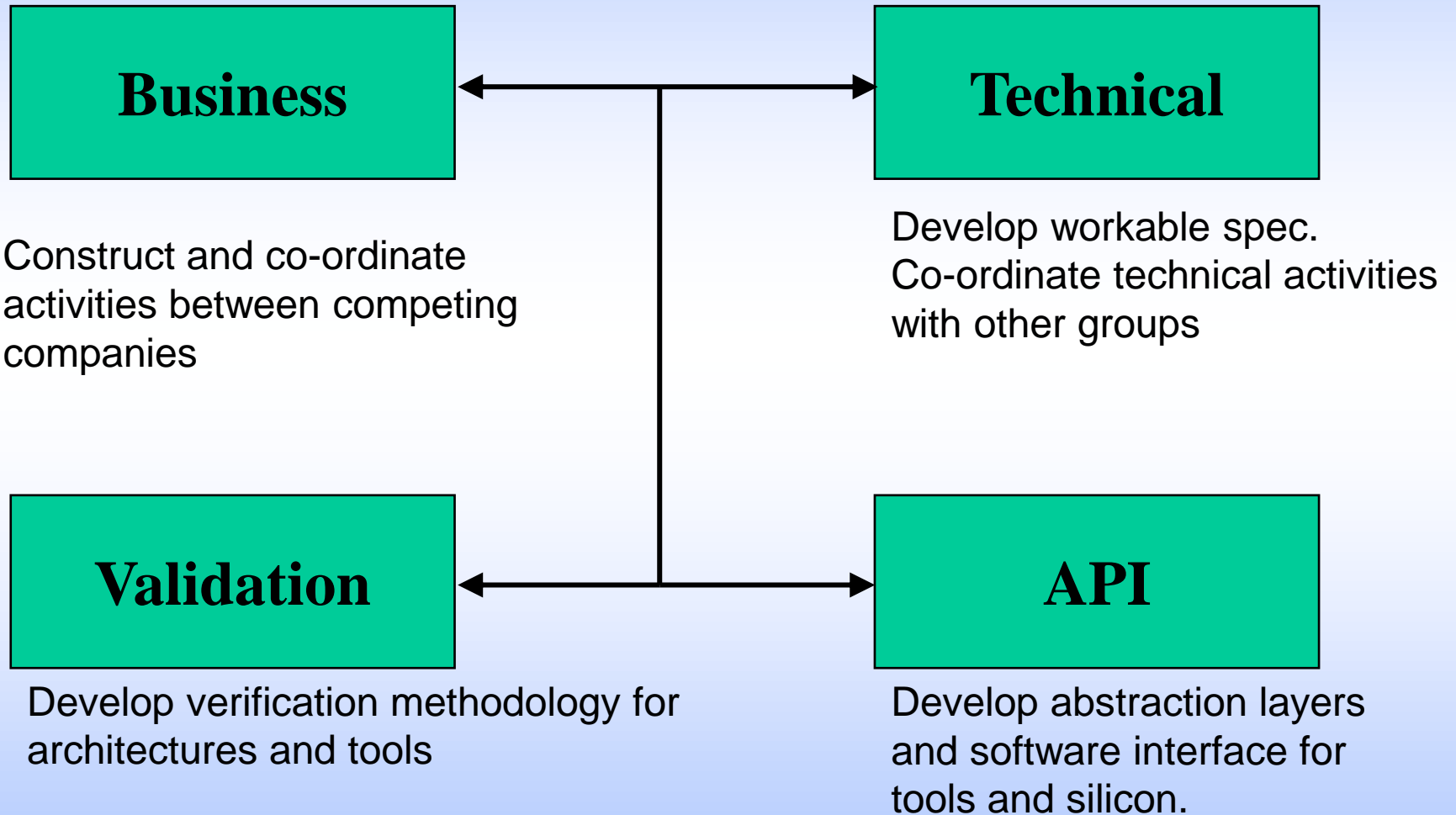
- ☑ Accurate Technologies
- ☑ [Agilent \(HP\)](#)
- ☑ Applied Dynamics Inc.
- ☑ Applied Microsystems Corp. (AMC)
- ☑ Ashling Microsystems
- ☑ Diab-SDS (WRS)
- ☑ Digital Logic Instruments (DLI)

Tool Vendors (cont'd):

- ☑ Emulation Technology
- ☑ [ETAS](#)
- ☑ Green Hills Software Inc. (GHS)
- ☑ Hitex Development Tools
- ☑ HIWARE
- ☑ Lauterbach Inc.
- ☑ Macraigor Systems, L.L.C.
- ☑ Metroworks Corporation
- ☑ Nohau
- ☑ Noral Micrologics
- ☑ SDS
- ☑ PLX Technology, Inc
- ☑ Tektronix
- ☑ Yokogawa Digital Computer Corp.

Nexus Committees and Their Roles

- Introduction
- **NEXUS Consortium**
- Development Features
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers



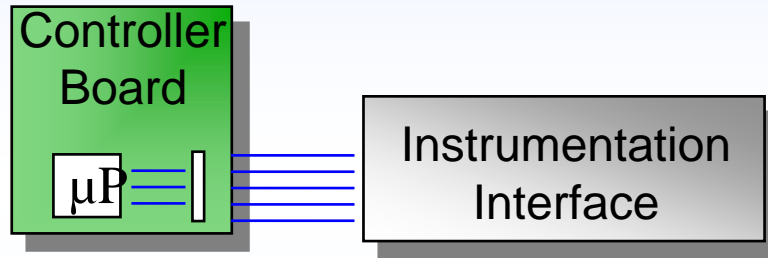
What Benefits Does Nexus Provide ?

- Introduction
- **NEXUS Consortium**
- Development Features
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers

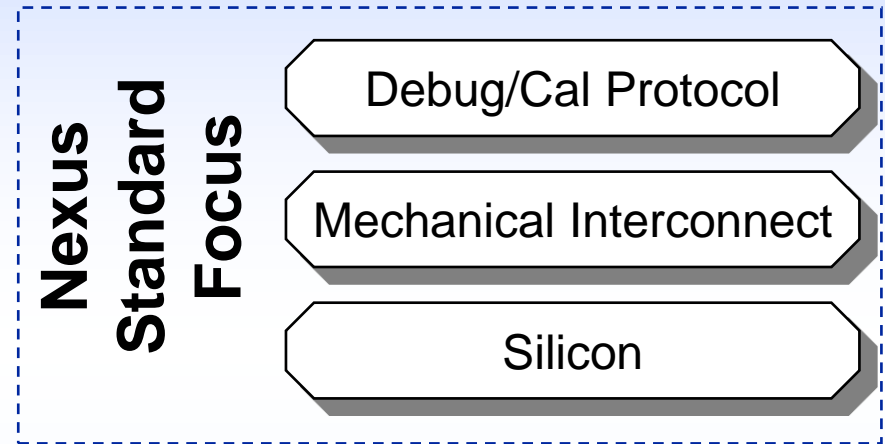
- Benefits for the developer:
 - Investment protection since the tools can be reused for another architecture
 - Tools can be available with the introduction of new embedded MCUs
 - Reduced customization, training and development time
- Benefits for the industry:
 - The verification of the tools & embedded MCU capabilities, prior to the use in customer applications, will reduce the risk & will improve the quality
- For the tools companies:
 - allows tool partner to focus investments at solving more customer problems and not at the various MCU traceport implementations
- For the semiconductor manufacturers:
 - It is much easier to get tool vendor support for new embedded MCUs.

Scope of IEEE-ISTO NEXUS 5001 Standard

- Introduction
- **NEXUS Consortium**
- Development Features
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers



- **Calibration**
- **Logic Analysis**
- **Run Control**
- **Rapid Prototyping**

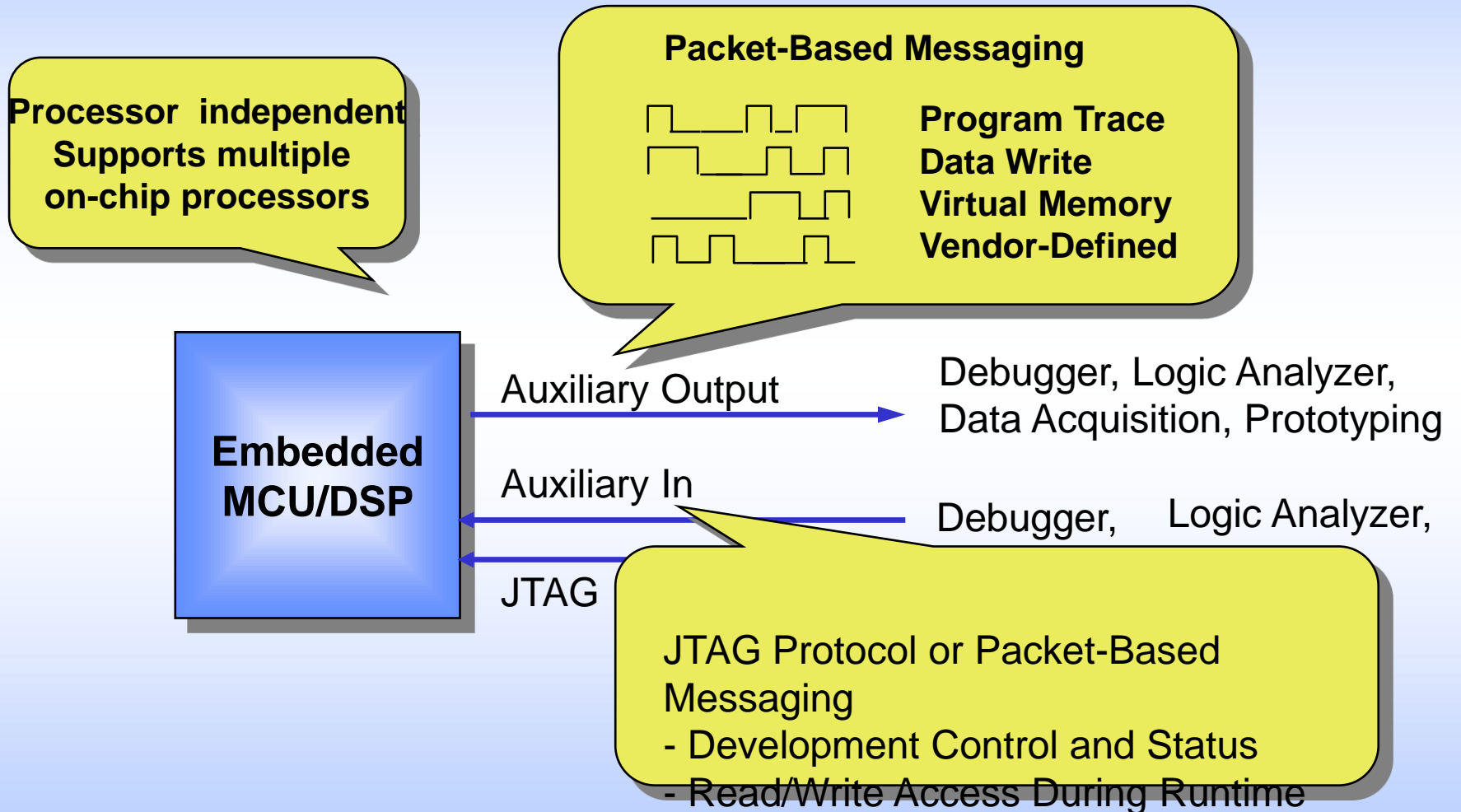


- Introduction
- NEXUS Consortium
- **Development Features**
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers

Development Features

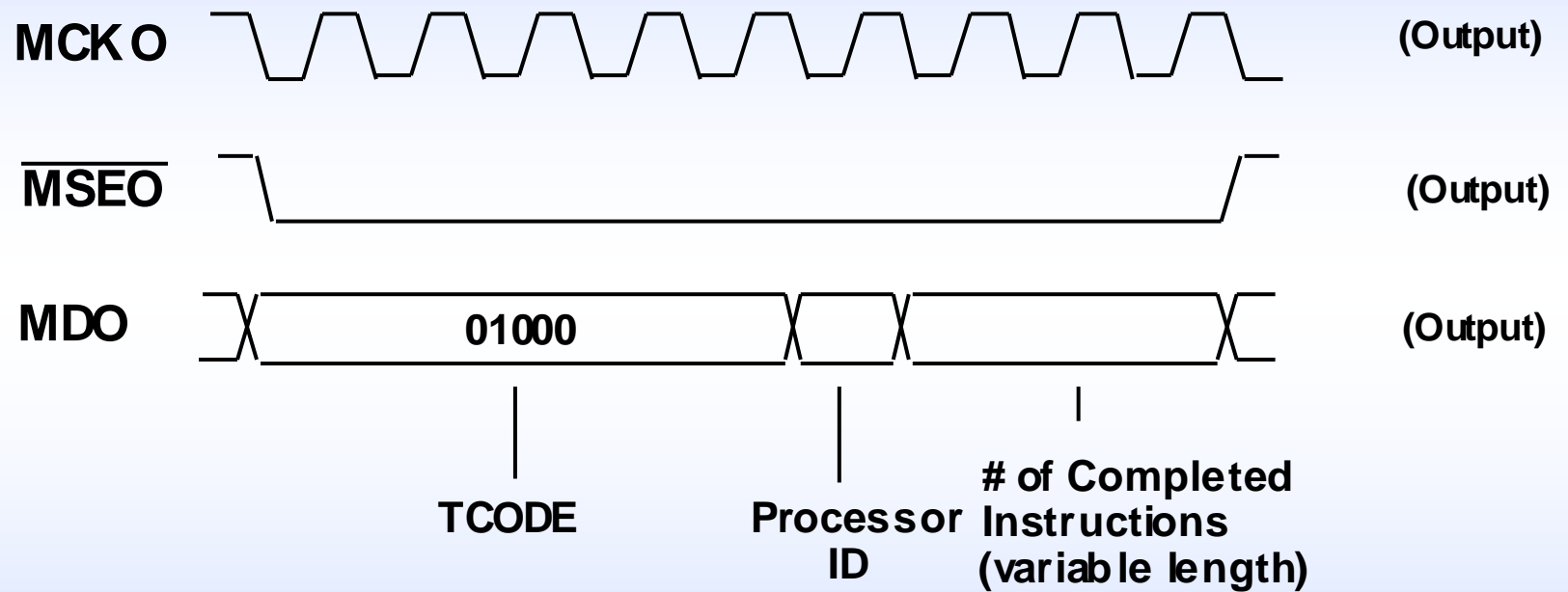
Nexus Development Features

- Introduction
- NEXUS Consortium
- **Development Features**
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers



Public Message - Direct Branch Message

- Introduction
- NEXUS Consortium
- **Development Features**
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers





NEXUS Compliance Classes

GEPDIS Class	Class 1	Class 2	Class 3	Class 4
Trace features	Trace not supported	Adds ownership trace and program trace via Auxiliary ports	Adds data write trace and read/write memory on the fly via Auxiliary ports	Allows tracing to be triggered by a watchpoint via Auxiliary ports
Debug communication method	Half-duplex communication (Limited bandwidth)	Communication may be full duplex using Auxiliary port (higher bandwidth)	Communication may be full duplex using Auxiliary port (higher bandwidth)	Communication may be full duplex using Auxiliary port (higher bandwidth)
run-time Control	Supports run time control features using JTAG interface	Supports run time control features using JTAG interface or Auxiliary port	Supports run time control features using JTAG interface or Auxiliary port	Supports run time control features using JTAG interface or Auxiliary port
Auxiliary Port Implementation	No Auxiliary Port	Allows Port sharing; the Auxiliary port may be shared with slow IO port pins (e.g. static Config pins that are latched at reset time).	Allows Port sharing with high-speed I/O ports.	Allows Port sharing with high-speed I/O ports.
Data acquisition	Not supported	Not supported	Supports data acquisition	Supports data acquisition
Memory Substitution	Not supported	Not supported	Not supported	Supports memory substitution (fetching or reading data over the GEPDIS auxiliary port) Triggering memory substitution on a watchpoint is an optional feature of Class 4 compliance

Static Development Features

- Introduction
- NEXUS Consortium
- **Development Features**
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers

Development Feature	Class 1	Class 2	Class 3	Class 4	Nexus Feature
STATIC DEVELOPMENT FEATURES					
Read/write user registers in debug mode	V	V	V	V	Refer to SECTION 5
Read/write user memory in debug mode	A	A	A	A	Read/Write Access
Enter a debug mode from reset	A	A	A	A	Development Control and Status
Enter a debug mode from user mode	A	A	A	A	
Exit a debug mode to user mode	A	A	A	A	
Single step instruction in user mode and re-enter debug mode	A	A	A	A	
Stop program execution on instruction/data breakpoint and enter debug mode (min. 2 breakpoints)	A	A	A	A	Breakpoints/Watch-points

V = Vendor specific implementation
A = NEXUS API support required

Dynamic Development Features

- Introduction
- NEXUS Consortium
- **Development Features**
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers

Development Feature	Class 1	Class 2	Class 3	Class 4	Nexus Feature
DYNAMIC DEVELOPMENT FEATURES					
Ability to set breakpoint or watchpoint	A	A	A	A	Breakpoints/Watchpoints
Device Identification	A	A & P	A & P	A & P	Device ID Message (see SECTION 6)
Ability to send out an event occurrence when watchpoint matches	P ¹	P	P	P	Watchpoint Message (see SECTION 6)
Monitor process ownership while processor runs in real-time		P	P	P	Ownership Trace
Monitor program flow while processor runs in real-time (logical address)		P	P	P	Program Trace
Monitor data writes while processor runs in real-time			P	P	Data Trace (Writes only)
Read/write memory locations while program runs in real-time			A & P	A & P	Read/Write Access
Program execution (instruction/data) from Nexus port for reset or exceptions				P	Memory Substitution
Ability to start ownership, program, or data trace upon watchpoint occurrence				A	Development Control and Status
Ability to start Memory Substitution upon watchpoint occurrence or upon program access of device-specific address				O	Development Control and Status
Monitor data reads while processor runs in real-time			O	O	Data Trace (Reads and Writes)
Low speed I/O port replacement and High speed I/O port sharing		O	O	O	Port Replacement/ Sharing

A = NEXUS API support required
P = NEXUS T-CODES protocol support required
O = optional but not required for Nexus Compliance

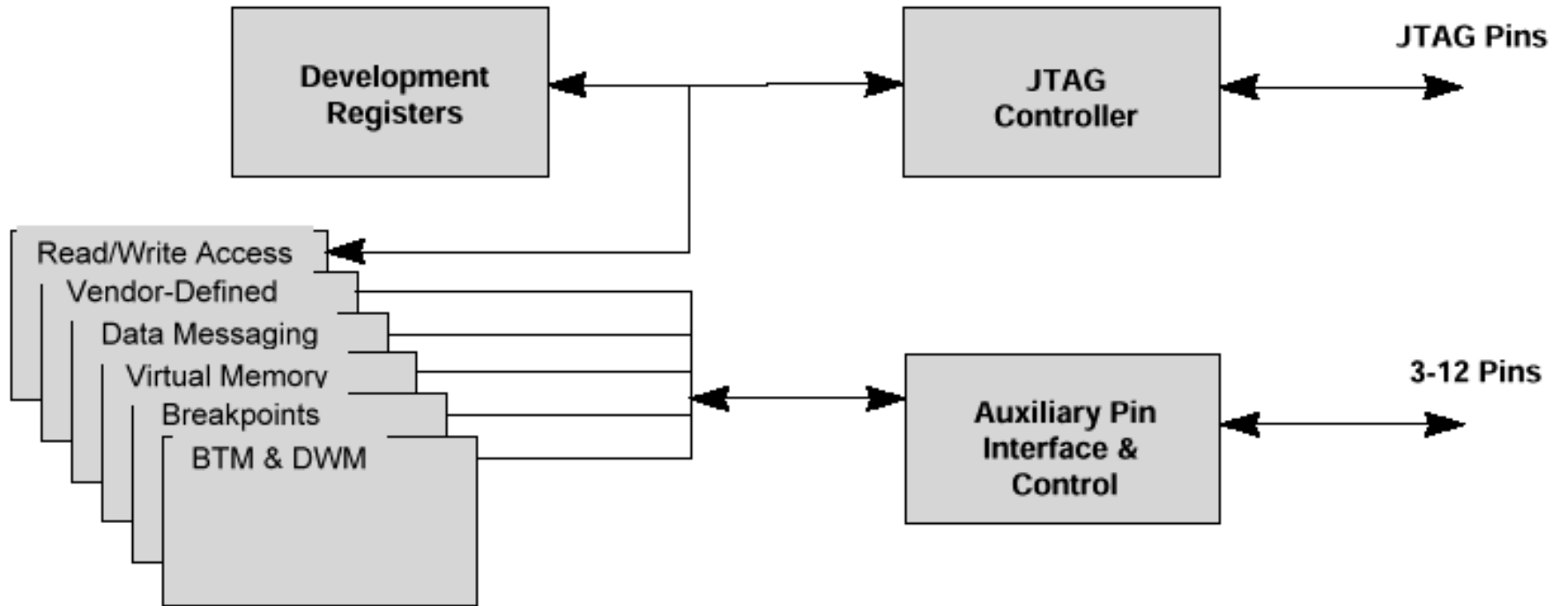
001 debug standard

- Introduction
- NEXUS Consortium
- Development Features
- **Hardware Interface**
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers

NEXUS On-Chip Hardware

NEXUS On-Chip Hardware

- Introduction
- NEXUS Consortium
- Development Features
- **Hardware Interface**
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers



- The hardware implementations will be dependent on the respective silicon vendor
- The design should be scalable, a Class 2 Implementation should be reusable for a Class 3 Implementation

Efficient Data Transfers using Nexus Hardware

- Introduction
- NEXUS Consortium
- Development Features
- **Hardware Interface**
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers

Example of how the target processor generates the address to send in a trace message:

Previous absolute address (A1) = 0x003FC01,
 Absolute address associated with new trace occurrence (A2) = 0x0003F365

A1 = 0000 0000 0000 0011 1111 1100 0000 0001
 A2 = 0000 0000 0000 0011 1111 0011 0110 0101
 A1 ⊕ A2 = 0000 0000 0000 0000 0000 1111 0110 0100

The unique portion of the address (M1), sent in the message (high-order zeros are suppressed):

M1 = 1111 0110 0100

Example of how the tool recreates the address based on its previously calculated address and the address contained in the trace message:

Previously calculated address (A1) = 0x003FC01,
 Address in message (M1) = 0xF64

A1 = 0000 0000 0000 0011 1111 1100 0000 0001
 M1 = 0000 0000 0000 0000 0000 1111 0110 0100
 A1 ⊕ M1 = 0000 0000 0000 0011 1111 0011 0110 0101

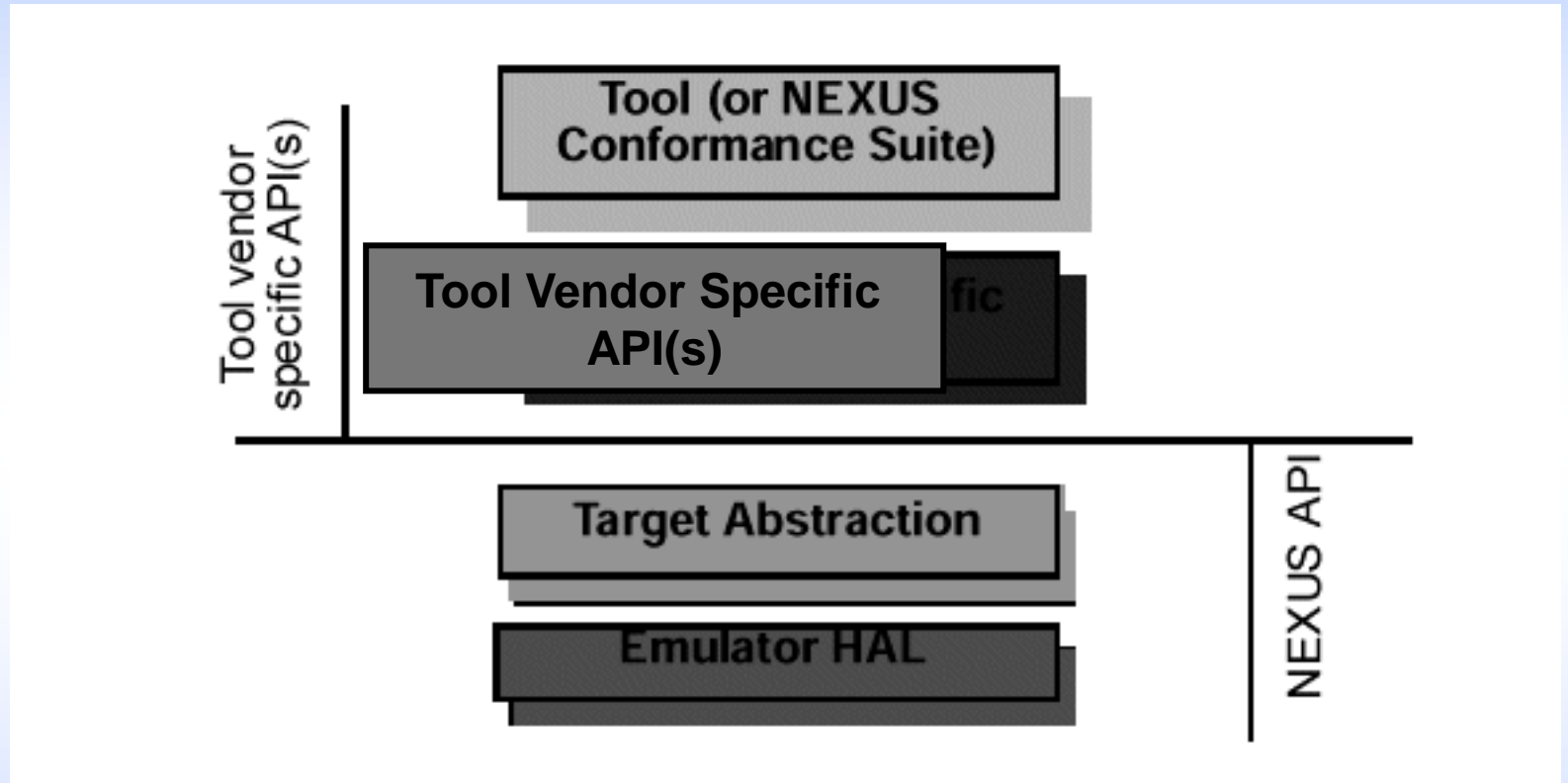
Address recreated by the tool = 0x0003F365

- Introduction
- NEXUS Consortium
- Development Features
- Hardware Interface
- **Software Interface**
- Board/Connector Interface
- Summary
- Question/Answers

NEXUS API Software Interface

NEXUS API Software Interface

- Introduction
- NEXUS Consortium
- Development Features
- Hardware Interface
- **Software Interface**
- Board/Connector Interface
- Summary
- Question/Answers



- Nexus Standard supplies API header files
- Semiconductor vendor provides Emulator vendor with Target Abstraction Layer
- Emulator vendor provides Tool vendor with integrated Nexus API layer

- Introduction
- NEXUS Consortium
- Development Features
- Hardware Interface
- **Software Interface**
- Board/Connector Interface
- Summary
- Question/Answers

Target Abstraction Layer

- Open - Establish connection to Target
- Close - Close connection to Target
- ioctl - Controlling connection to Target
- ReadMem / WriteMem - Memory access to Target
- SetEvent / GetEvent / ClearEvent - Event handling of Target
- GetLastError - Error Handling/Recovery

Emulator Hardware Abstraction Layer (HAL)

- Open - Establish connection to Emulator
- Close - Close connection to Emulator
- ReadNJR / WriteNJR - Access of JTAG resources via Emulator.

- Introduction
- NEXUS Consortium
- Development Features
- Hardware Interface
- Software Interface
- **Board/Connector Interface**
- Summary
- Question/Answers

Board / Connector Interface

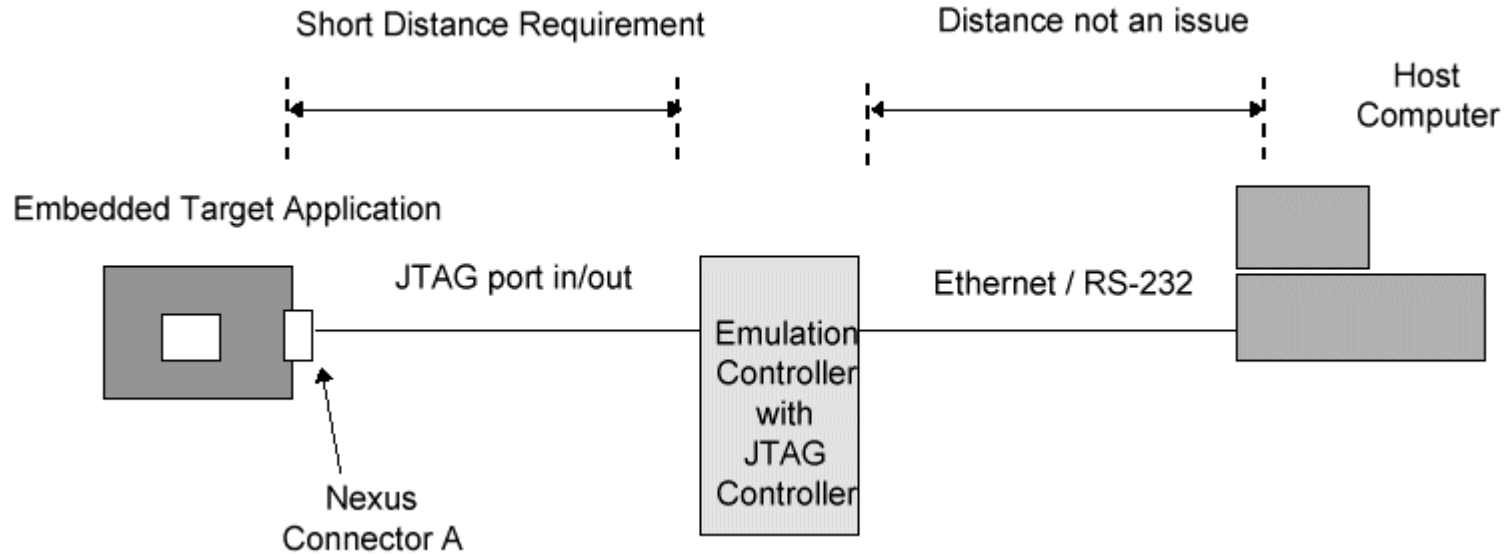
Nexus Connector Interface

- Introduction
- NEXUS Consortium
- Development Features
- Hardware Interface
- Software Interface
- **Board/Connector Interface**
- Summary
- Question/Answers

Pin Name	Conn. A JTAG Port only	Conn. B Option 1 - JTAG	Conn. B Option 2 - Aux Port	Conn. B Option 3 - Mixed	Conn. C Aux Port Only	Comments
MCKI			1		1	Auxiliary Port
MDI			2		4	
MSEI			1		1	
MCKO			1	1	1	
MDO			4	2	8	
MSEO			1	1	2	
EVTO	1	1	1	1	1	
EVTI	1	1	1	1	1	
RSTI			1		1	
PORT					16	Port Replacement
JTAG	5	5		5		IEEE 1149.1
RDY	1	1		1		
Vendor Defined	1	1	1	1	2	
VREF	1	1	1	1	1	System Signals
RESET	1	1	1	1	1	
CLOCKOUT	1	1				
Other					2	

NEXUS Connector A

- Introduction
- NEXUS Consortium
- Development Features
- Hardware Interface
- Software Interface
- **Board/Connector Interface**
- Summary
- Question/Answers



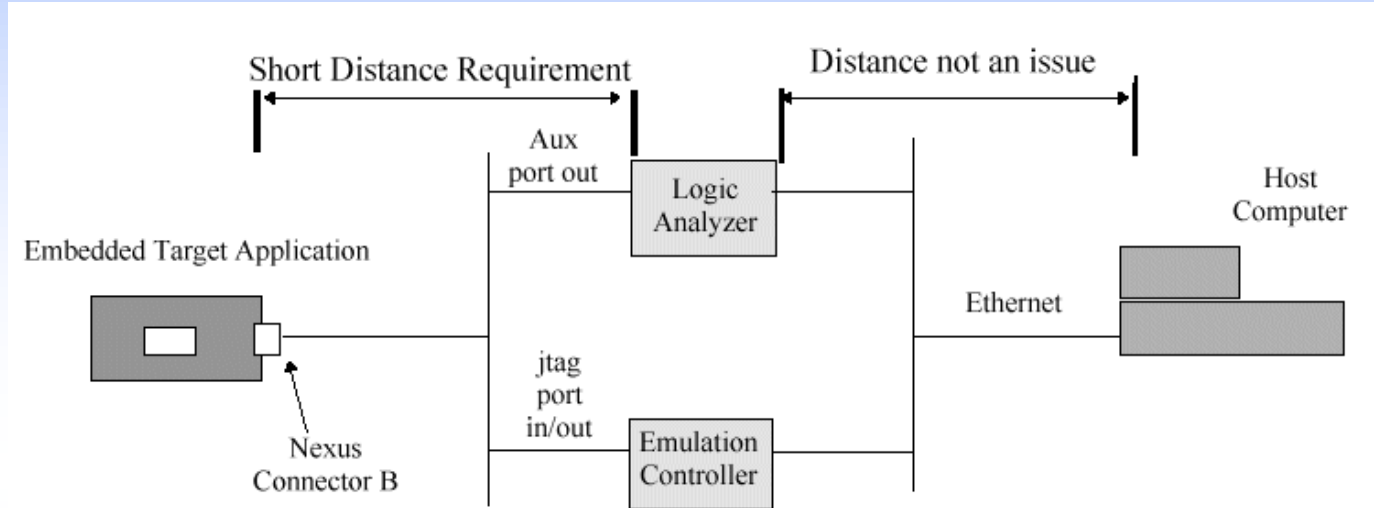
- **20 Pin Connector which supports JTAG 5 pin Protocol**
- **Amp 2 row x 10 columns,**
- **0.050 inch pitch pin spacing and 0.10 in row spacing.**
- **Includes System_Clock_out, System_Reset_in,**
- **Data_Rdy_out, Evti_in, Evto_out**
- **Supports 3.3v and lower with System_Vref_out**

Signal Name	I/O	Pin	Pin	I/O	Signal Name
RESET	IN	1	2	OUT	VREF
EVTI*	IN	3	4	-	GND
TRST*	IN	5	6	-	GND
TMS	IN	7	8	-	GND
TDI	IN	9	10	-	GND
TCK	IN	11	12	-	GND
TDO	OUT	13	14	-	GND
CLOCKOUT*	OUT	15	16	-	GND
EVTO*	OUT	17	18	-	GND
RDY*	OUT	19	20	I or O	Vendor Defined*

IEEE-ISTO NEXUS 5001 debug standard

NEXUS Connector B

- Introduction
- NEXUS Consortium
- Development Features
- Hardware Interface
- Software Interface
- Board/Connector Interface
- Summary
- Question/Answers

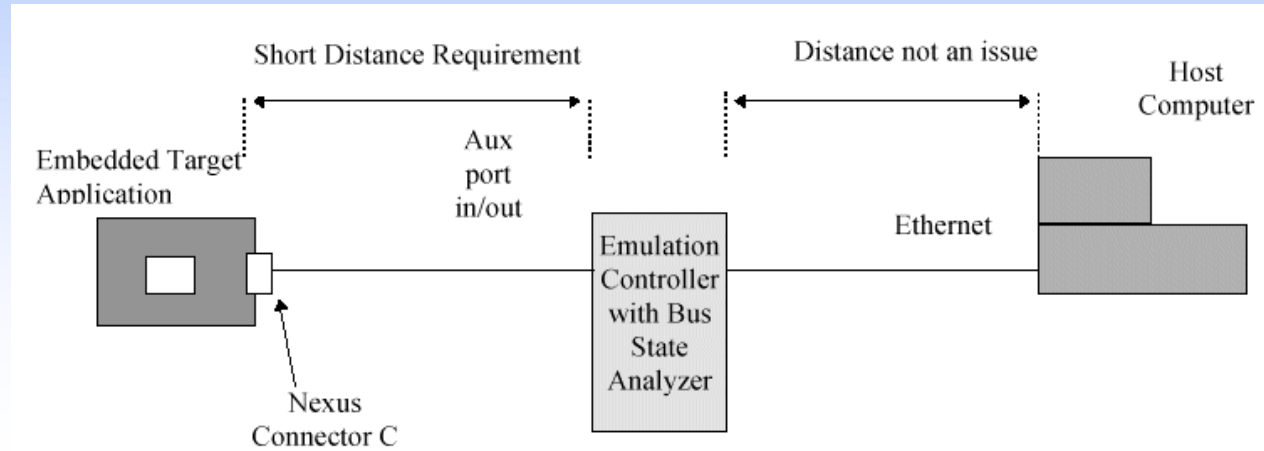


- 30 Pin Amp Connector supports JTAG 5 pin Protocol and/or Aux Port
- Amp 3 row x 10 columns, 0.050 in. column X 0.10 in row spacing

Signal Name	Signal Name	Signal Name	I/O	Pin	Pin	I/O	Signal Name
1) JTAG Mode	2) Aux Mode	3) Mixed Mode					
RESET	RESET	RESET	IN	1	2	OUT	VREF
EVTI*	EVTI*	EVTI*	IN	3	4	-	GND
TRST*	RSTI	TRST*	IN	5	6	-	GND
TMS	RESERVED	TMS	IN	7	8	-	GND
RESERVED	MDI1*	RESERVED	IN	9	10	-	GND
TDI	MDI0	TDI	IN	11	12	-	GND
TCK	MCKI	TCK	IN	13	14	-	GND
RESERVED	MSEI	RESERVED	IN	15	16	-	GND
TDO	MDO3*	TDO	OUT	17	18	-	GND
RDY*	MDO2*	RDY*	OUT	19	20	-	GND
RESERVED	MDO1*	MDO1*	OUT	21	22	-	GND
RESERVED	MDO0	MDO0	OUT	23	24	-	GND
CLOCKOUT*	MCKO	MCKO	OUT	25	26	-	GND
RESERVED	MSEO	MSEO	OUT	27	28	-	GND
EVTO*	EVTO*	EVTO*	OUT	29	30	I or O	Vendor Defined*

NEXUS Connector C

- Introduction
- NEXUS Consortium
- Development Features
- Hardware Interface
- Software Interface
- **Board/Connector Interface**
- Summary
- Question/Answers



- Intended for target processors which support wide aux. port and port replacement requirements.
- 80 Pin Samtec MOLC/FOLC Connector
- Amp 2 row x 10 columns, 0.050 in. column X 0.050 in row spacing

Signal Name	I/O	Pin	Pin	Pin	Pin	I/O	Signal Name
RESET	IN	1	2-UBATT*	3	4	OUT	VREF
EVTI*	IN	5	6	7	8	OUT	EVTO*
RSTI	IN	9	10	11	12	I/O	Vendor Defined*
MDI3*	IN	13	14	15	16	I/O	Vendor Defined*
MDI2*	IN	17	18	19	20	I/O	PORT15*
MDI1*	IN	21	22	23	24	I/O	PORT14*
MDI0	IN	25	26	27	28	I/O	PORT13*
MCKI	IN	29	30	31	32	I/O	PORT12*
MSEI	IN	33	34	35	36	I/O	PORT11*
MDO7*	OUT	37	38	39	40	I/O	PORT10*
MDO6*	OUT	41	42	43	44	I/O	PORT9*
MDO5*	OUT	45	46	47	48	I/O	PORT8*
MDO4*	OUT	49	50	51	52	I/O	PORT7*
MDO3*	OUT	53	54	55	56	I/O	PORT6*
MDO2*	OUT	57	58	59	60	I/O	PORT5*
MDO1*	OUT	61	62	63	64	I/O	PORT4*
MDO0	OUT	65	66	67	68	I/O	PORT3*
MCKO	OUT	69	70	71	72	I/O	PORT2*
MSEO1*	OUT	73	74	75	76	I/O	PORT1*
MSEO0	OUT	77	78	79-UBATT*	80	I/O	PORT0*

- Introduction
- NEXUS Consortium
- Development Features
- Hardware Interface
- Software Interface
- Board/Connector Interface
- **Summary**
- Question/Answers

Global Embedded Processor Debug Interface Standard

Eliminating Development Problems so that Real Application Problems can be Solved

- Introduction
- NEXUS Consortium
- Development Features
- Hardware Interface
- Software Interface
- Board/Connector Interface
- **Summary**
- Question/Answers

**For a copy of the current specification
and to provide technical feedback
visit our web site at**

www. www.ieee-isto.org/Nexus5001/

Questions?

