An Introduction to the **MPC57xx Nexus Aurora Interface**

FTF-AUT-F0344

Randy Dees

A P R . 2 0 1 4
 Agenda

• MPC57xx Debug Overview
• Nexus Aurora Trace Overview
• Board and Connector Recommendations
• MPC57xx Emulation Devices
• MPC57xx Trace Adapters
• Trace Tool Vendor Support
• Summary
Objective

- Some of the MPC57xx devices include a high-speed Nexus trace port that is built on the Xilinx® Aurora interface
- This session reviews the following:
  - Introduction to the Automotive Nexus Aurora Trace interface
  - Board-level hardware requirements for the multi-lane 1.25 Gb/s interface, including capabilities for trace information from the MCU
  - An overview of the internal features and construction of the MPC57xx emulation devices
  - An overview of trace adapters available for some devices
MPC57xx Debug Overview
MPC57xx Debug: Features and Advantages

- Rich set of debug and calibration features:
  - High speed Nexus trace port via Aurora (1.25 Gbps per lane, 2 or 4 lanes) for high bandwidth trace capability
- **Standard Nexus Client** for e200zx cores, eDMA, and timer subsystems
- JTAG 1149.1 based debug architecture for run control
- JTAG 1149.7 support
  - Two-pin interface allows simplified routing and minimized pin count, while keeping all 1149.1 style features
- Trace to on-chip memory on both the production device and the on the emulation devices for family members that support emulation devices
  - 16K SRAM on the production devices can be used for overlay or trace
  - 1M or 2M overlay RAM when using emulation device that can be partitioned between trace and overlay
- **Sequence Processing Unit**
  - Similar to an on-chip logic analyzer triggering, provides advanced debug, watch-point, breakpoint features for static debug or for limiting trace information
IEEE-ISTO 5001 Nexus Classes (MPC57xx support)

Class 1
Run Time Control
- Start/stop code execution
- Read/write MCU registers / memory
- Breakpoints
- Single step instructions
- Read Nexus device ID

Class 2
Dynamic Debug
- Real-time process/task ownership tracing
- Trigger a nexus message on an event
- Real time, non-intrusive instruction trace

Class 3
Data Trace
- Real-time access of registers / memory (read/write)
- Real-time data trace (writes)
Optional features supported:
- Real-time data trace (reads)
- Data acquisition

Class 4
Advanced Debug
- Watchpoint triggered trace event
- Message over-run control
Nexus Class Definition – Class 1

Class 1
- Run Time Control
  - Read/write MCU registers / memory
  - Set / clear breakpoints
  - Stop / start code execution
  - Control entry into / exit from debug mode (from reset and user modes)
  - Stop execution on hitting a breakpoint and enter debug mode
  - Single step instructions
  - Read Nexus device ID

Class 2
- Dynamic Debug

Class 3
- Data Trace

Class 4
- Advanced Debug
Nexus Class Definition – Class 2

Class 1
Run Time Control

Class 2
Dynamic Debug

Class 3
Data Trace

Class 4
Advanced Debug

Class 2
All Class 1 features plus:

- **Ownership trace messages** – Real time process / task ownership tracing
- **Watchpoint messaging** – Trigger a Nexus message on an event
- **Program trace messages** – Real-time, non-intrusive instruction trace

Optional Features:
- **Port Replacement** (of slow GPIO)
Nexus Class Definition – Class 3

Features that Freescale supports on many automotive devices that support only Class 2

Class 1
Run Time Control

Class 2
Dynamic Debug

Class 3
Data Trace

Class 3
All Class 2 features plus:

Real time data access – Registers / memory can be read/written in real time
  • Real time data trace (WRITES)

Optional Features:
  • Real time data trace (READS)
  • Transmission of additional data used for data acquisition
Nexus Class Definition – Class 4

Features that Freescale supports on many automotive devices that support only Class 2 or 3

Class 1
Run Time Control

Class 2
Dynamic Debug

Class 3
Data Trace

Class 4
Advanced Debug

Class 4
All Class 3 features plus:

- **Watchpoint triggering** – Allows a watchpoint to trigger trace event

- **Memory substitution** – MCU can run code from memory in development tool (ROM emulation)

- **Over-run control** – Allows Nexus to stop core if buffers will overflow

Optional Features:

- Start memory substitution on watchpoint
Trace Options Overview

- MPC55xx, MPC56xx, and some MPC57xx devices
  - No trace capability in the production package (primarily low pin-count packages and devices)
    ▪ Trace adapters available to fit production footprint
  - Nexus Parallel Trace port
    ▪ 4 to 16 bits of message data outputs
    ▪ 1 or 2 pin message start/end outputs + Nexus clock

- MPC57xx extra options on some devices
  - Nexus trace to memory
  - Nexus high speed Aurora trace
    ▪ 2 or 4 lanes
Multicore Trace

Program, data, and ownership trace, plus watchpoint messaging, and data acquisition provide a multitude of development capabilities.

- **Code coverage**
  - Do all lines of code get executed?
  - Are all conditional branch paths executed?

- **Execution time**
  - Task, function execution times

- **Code profiling**
  - What functions are called the most?
  - What functions consume the most time?

- **Task analysis**
  - What tasks are run, how often

- **Cache coverage**
  - Analysis of program/data caching to optimize performance

- **Core loading**
  - What is the loading of each core?
  - Can programs be segregated differently to improve the balance of performance?
### MPC57xx Nexus Interfaces

<table>
<thead>
<tr>
<th>High-speed Serial Nexus (Aurora)</th>
<th>2/4 Lanes</th>
</tr>
</thead>
<tbody>
<tr>
<td>• MPC5777M ED</td>
<td></td>
</tr>
<tr>
<td>• MPC5746M ED</td>
<td></td>
</tr>
<tr>
<td>• MPC5746R ED</td>
<td></td>
</tr>
<tr>
<td>• MPC5775K</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>High-speed Serial Nexus (Aurora) + Parallel Nexus</th>
<th>4 MDO/2 Lanes</th>
</tr>
</thead>
<tbody>
<tr>
<td>• MPC5744P</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parallel Trace</th>
<th>4-16 MDO</th>
</tr>
</thead>
<tbody>
<tr>
<td>• MPC5777C</td>
<td></td>
</tr>
<tr>
<td>• MPC574xG/C</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Trace to Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>• MPC5777M</td>
</tr>
<tr>
<td>• MPC5746M</td>
</tr>
<tr>
<td>• MPC5746R</td>
</tr>
</tbody>
</table>
MPC57xx Nexus Parallel and Serial Trace Support

- Some MPC57xx devices support both Nexus parallel and serial (Aurora) trace interface
- All e200 cores on these devices support Class 3+
- Trace to memory is not supported

<table>
<thead>
<tr>
<th>Device</th>
<th>Nexus parallel trace</th>
<th>Nexus serial (Aurora) trace</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPC5744P</td>
<td>4¹</td>
<td>2²</td>
</tr>
<tr>
<td>MPC5775K</td>
<td>16³</td>
<td>4</td>
</tr>
</tbody>
</table>

1. Available in all packages
2. 257 MAPBGA package only
3. Internally, 16 parallel Nexus Message Data Outputs are available, however, these are not available in any of the standard packages.
MPC57xx Nexus Serial (only) Trace Support

- Some MPC57xx devices only support the Nexus serial (Aurora) trace interface on Emulation Devices (ED)
- All e200 cores on these devices support Class 3+
- Trace to memory is also supported

<table>
<thead>
<tr>
<th>Device</th>
<th>Nexus serial (Aurora) trace</th>
<th>Nexus trace to memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPC5746M ED</td>
<td>4</td>
<td>1 MB</td>
</tr>
<tr>
<td>MPC5746R ED</td>
<td>4</td>
<td>1 MB</td>
</tr>
<tr>
<td>MPC5777M ED</td>
<td>4</td>
<td>2 MB</td>
</tr>
</tbody>
</table>
MPC57xx Nexus Trace to Memory Support

- Some MPC57xx devices only support Nexus trace to memory on the production devices
- All e200 cores on these devices support Class 3+
- These devices also support the Sequence Processing Unit to narrow the scope of information that is traced

<table>
<thead>
<tr>
<th>Device</th>
<th>Nexus trace to memory size</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPC5746M</td>
<td>16 KB</td>
</tr>
<tr>
<td>MPC5746R</td>
<td>16 KB</td>
</tr>
<tr>
<td>MPC5777M</td>
<td>16 KB</td>
</tr>
</tbody>
</table>
Nexus Aurora Trace Overview
Nexus Trace Overview

- Nexus uses Message Data Outputs (MDO) and Message Start/End Outputs (MSEO) to transmit trace information from a device to the external word.

- For a Nexus parallel auxiliary (output) port, there can be 4 to 16 MDO signals and either 1 or 2 MSEO signals.

- Nexus Serial (Aurora-based) trace internally implements 30 MDO signals and 2 MSEO signals.
  - The Nexus Aurora interface takes these 32 Nexus signals and formats them into the Aurora high-speed serial protocol for transmitting out of the device.
  - Aurora was developed by Xilinx for taking parallel information and transmitting it over a variable number of serial lanes. The Aurora protocol “stripes” the data across multiple lanes (LVDS pair).
What is Aurora?

The Aurora Standard
A scalable, high-speed serial, link-level interface
- Common protocol for single and multi lane channels
  - Serial full duplex or serial simplex operation
  - System-synchronous or asynchronous operation
  - Arbitrary data transfers: packets or words
- Optional flow control & expedited messaging
- 8B/10B data encoding

Freescale Automotive Implementation
Simplex
- Tool synchronous
- 2 or 4 lanes
- 625M baud to 1.25G baud per lane
- 8B/10B data encoding
Aurora Protocol

- Aurora transmits continuous data. If there are no Nexus messages, Idle messages are transmitted
- Clock Compensation (CC) symbols are transmitted periodically
- User PDUs are the actual Protocol Data Units (Nexus trace messages)
Nexus Aurora PDU (Message Formatting)

- Nexus messages are formatted as a 30-bit Message Data Output, plus 2 Message Start/End Signals
- This 32-bit data is consolidated by the Nexus Aurora Router from each of the Nexus clients

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reset</td>
<td>*1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th>27</th>
<th>28</th>
<th>29</th>
<th>30</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>MDO[29:14]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reset</td>
<td>*1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>32</th>
<th>33</th>
<th>34</th>
<th>35</th>
<th>36</th>
<th>37</th>
<th>38</th>
<th>39</th>
<th>40</th>
<th>41</th>
<th>42</th>
<th>43</th>
<th>44</th>
<th>45</th>
<th>46</th>
<th>47</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>MDO[13:0]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reset</td>
<td>*1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MSEO [1:0]
Aurora Lane Striping

• Aurora is a protocol for taking parallel data and serializing it over a LVDS connection
• The Aurora protocol handles dividing the incoming parallel data into different lanes
Nexus Aurora Lane Formatting

- This example shows how the 32-bit Nexus words are striped across multiple lanes.
- Note the lane order is little endian.
Aurora Transmit Signals (1.25 GHz)

Transmit Eye diagram measured at the termination resistor (no stubs)

Transmit Eye diagram measured with Freescale Aurora breakout board
Nexus Messages

- Typical start-up showing the Nexus device identification messages

<table>
<thead>
<tr>
<th>TCODE=00 SRC=0 DSM STATUS=0000</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCODE=01 DIDM DID=0AF8101D</td>
</tr>
<tr>
<td>TCODE=01 DIDM DID=0AF0A01D</td>
</tr>
<tr>
<td>TCODE=08 SRC=F ETYPE=C EVCODE=00</td>
</tr>
<tr>
<td>TCODE=00 SRC=2 DSM STATUS=0000</td>
</tr>
<tr>
<td>TCODE=09 SRC=2 PT-SM MAP=0 ICNT=0000 F-ADDR=01002111</td>
</tr>
<tr>
<td>TCODE=08 SRC=E ETYPE=0 EVCODE=03</td>
</tr>
<tr>
<td>TCODE=03 SRC=2 PT-DBM ICNT=0004</td>
</tr>
<tr>
<td>TCODE=03 SRC=2 PT-DBM ICNT=000C</td>
</tr>
<tr>
<td>TCODE=03 SRC=2 PT-DBM ICNT=0005</td>
</tr>
</tbody>
</table>
Trace Reconstruction Example

Blue line is the same point in all windows. Upper left – chart view, lower left is the Nexus trace messages, and the right is the reconstructed program flow.
Board and Connector Recommendations
Aurora Signal Terminations

- Debugger terminations typically included on chip (e.g. Xilinx FPGA)
- MCU input decoupling capacitors should be located close to device
Nexus Aurora Layout Guidelines

Care should be taken when laying out the Nexus Aurora signals on the printed circuit board.

- **Connector placement**
  Even side parallel to the edge of the board.
  Odd side nearest to the MCU.

- **Printed circuit board**
  Controlled impedance highly recommended

- **Routing**
  Route LVDS pairs parallel to and near each other.
  Use 2 vias maximum.

- **Keep-out areas**
  Tools may have a specific keep-out area requirement

- **Signal return path**
  Nexus Aurora signals should be placed 1 dielectric away from a solid ground.

- **Routing**
  All transmit pairs should be approximately the same length.
  Clock can be a different length than the transmit signals.
  Keep necking to less than 0.1 inch to avoid discontinuities.
# MPC57xx Nexus Connector Definition

<table>
<thead>
<tr>
<th>Position</th>
<th>Signal</th>
<th>Pin Number</th>
<th>Pin Number</th>
<th>Signal</th>
<th>Nexus signal name</th>
</tr>
</thead>
<tbody>
<tr>
<td>GND</td>
<td>Latch</td>
<td>GND</td>
<td></td>
<td>VREF</td>
<td>VREF</td>
</tr>
<tr>
<td>TX0+</td>
<td>1</td>
<td>2</td>
<td>VREF</td>
<td>VREF</td>
<td></td>
</tr>
<tr>
<td>TX0-</td>
<td>3</td>
<td>4</td>
<td>TCK</td>
<td>TCK</td>
<td></td>
</tr>
<tr>
<td>GND</td>
<td>5</td>
<td>6</td>
<td>TMS</td>
<td>TMS</td>
<td></td>
</tr>
<tr>
<td>TX1+</td>
<td>7</td>
<td>8</td>
<td>TDI</td>
<td>TDI</td>
<td></td>
</tr>
<tr>
<td>TX1-</td>
<td>9</td>
<td>10</td>
<td>TDI</td>
<td>TDI</td>
<td></td>
</tr>
<tr>
<td>GND</td>
<td>11</td>
<td>12</td>
<td>TRST</td>
<td>JCOMP</td>
<td></td>
</tr>
<tr>
<td>TX2+</td>
<td>13</td>
<td>14</td>
<td>GEN_IO0</td>
<td>EVT1</td>
<td></td>
</tr>
<tr>
<td>TX2-</td>
<td>15</td>
<td>16</td>
<td>EVTO</td>
<td>EVT1</td>
<td>EVTO(0)</td>
</tr>
<tr>
<td>GND</td>
<td>17</td>
<td>18</td>
<td>EVT0(0)</td>
<td>EVT0</td>
<td>EVTO</td>
</tr>
<tr>
<td>TX3+</td>
<td>19</td>
<td>20</td>
<td>GEN_IO3</td>
<td>RSTOUT</td>
<td>GEN_IO3</td>
</tr>
<tr>
<td>TX3-</td>
<td>21</td>
<td>22</td>
<td>RESET</td>
<td>RESET</td>
<td></td>
</tr>
<tr>
<td>GND</td>
<td>23</td>
<td>24</td>
<td>GND</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>(TX4+)</td>
<td>25</td>
<td>26</td>
<td>CLK+</td>
<td>CLK+</td>
<td></td>
</tr>
<tr>
<td>(TX4-)</td>
<td>27</td>
<td>28</td>
<td>CLK-</td>
<td>CLK-</td>
<td></td>
</tr>
<tr>
<td>GND</td>
<td>29</td>
<td>30</td>
<td>GND</td>
<td>GND</td>
<td></td>
</tr>
<tr>
<td>(TX5+)</td>
<td>31</td>
<td>32</td>
<td>EVTO1/RDY</td>
<td>WDT</td>
<td>GEN_IO5</td>
</tr>
<tr>
<td>(TX5-)</td>
<td>33</td>
<td>34</td>
<td>RDY</td>
<td>WDT</td>
<td>GEN_IO5</td>
</tr>
</tbody>
</table>
Emulation Device

• Some devices will support an emulation device that includes the production die plus a companion (buddy) die
• The buddy die (BD) includes
  - JTAG Interface
  - 1 to 2 MB of calibration/overlay SRAM (can be divided)
  - Nexus Aurora physical interface
  - Nexus read/write client for the buddy SRAM (BD is accessible when power to the PD is off)
  - Data write processing unit (some devices) for filtering data trace messages

• Advantages
  - Actual production die is used so the functionality is the same
  - Lower production device cost
  - Supports large “on-chip” SRAM for trace or calibration overlay (1MB - 2 MB)
  - High-speed physical trace interface
  - Separate power supply for loading calibration memory with production die powered off

• Disadvantages
  - No physical trace interface on production device to tools
  - Limited on-chip trace capability for production device (16K)
**JTAG Access of Debug Resources**

- **IEEE 1149.1** or **IEEE 1149.7**

- Control of the JTAG interface is passed from the top level JTAG controller (in the Debug and Calibration Interface) to other JTAG clients with ACCES_AUX commands.

- Note: the PAUSE-DR/UPDATE-DR always returns the control of the JTAG TAP to the top level JTAG controller.

- IDCODE of both DCI/JTAGC modules can be read to determine the exact device and emulation device.

---

**MPC5746M Emulation Device shown**
Nexus Trace Message Flow

MPC5746M Emulation Device shown
MPC5746M Emulation Device

- Logic on the buddy connects directly to logic on PD
- Active pad on buddy connects to bond wire without production silicon
Copper Pillar, Chip-on-Chip Package Construction
Trace Adapters
MPC57xx Trace Adapters

• For some products, Freescale will make trace adapters available
• These trace adapters give access to the debug pins (JTAG and the Nexus trace pins), but fit onto a smaller footprint package. These were available for some of the MPC560xB products and will be available for some of the MPC57xx devices including the MPC57446R.
• The MPC5746R trace adapter will adapt the 292 MAPBGA package to the 252 MAPBGA or 176 LQFP packages footprint in the target system. This is shown below.

<table>
<thead>
<tr>
<th>Trace Adapter Support</th>
<th>Footprints</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPC5746R</td>
<td>1144 QFP, 176 QFP, 252 BGA</td>
</tr>
<tr>
<td>MPC574xG</td>
<td>176 (216) QFP, 292 BGA</td>
</tr>
<tr>
<td>MPC5746M</td>
<td>416 BGA, 516 BGA</td>
</tr>
<tr>
<td>MPC5777M</td>
<td></td>
</tr>
</tbody>
</table>
MPC5746R 252MAPBGA Trace Adapter Concept

- 292 MAPBGA package mapped into a 252 BGA footprint
Standard JTAG Connector
(modified for new future signals)

Based primarily on the existing MPC55xx/MPC56xx 14-pin JTAG connector.

<table>
<thead>
<tr>
<th>JTAG Function</th>
<th>Direction (from MCU)</th>
<th>Pin</th>
<th>Pin</th>
<th>Direction (from MCU)</th>
<th>JTAG Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>TDI</td>
<td>In</td>
<td>1</td>
<td>2</td>
<td>-</td>
<td>GND</td>
</tr>
<tr>
<td>TDO</td>
<td>Out</td>
<td>3</td>
<td>4</td>
<td>-</td>
<td>GND</td>
</tr>
<tr>
<td>TCK (TCKC)</td>
<td>In</td>
<td>5</td>
<td>6</td>
<td>-</td>
<td>GND</td>
</tr>
<tr>
<td>/EVT¹</td>
<td>In/Out</td>
<td>7</td>
<td>8</td>
<td>In</td>
<td>/PORST²</td>
</tr>
<tr>
<td>/RESET or /ESR0</td>
<td>In/Out</td>
<td>9</td>
<td>10</td>
<td>In/Out</td>
<td>TMS (TMSC)</td>
</tr>
<tr>
<td>VREF</td>
<td>-</td>
<td>11</td>
<td>12</td>
<td>-</td>
<td>GND</td>
</tr>
<tr>
<td>/RDY (EVT)³</td>
<td>Out (In)</td>
<td>13</td>
<td>14</td>
<td>In</td>
<td>JCOMP</td>
</tr>
</tbody>
</table>

1. Previously this was /EVTI, now EVTI or EVTO (EVTI preferred, programmable on some devices)
2. Previously this pin was a no connect on the MPC55xx and MPC56xx devices
3. RDY not available on all devices, use for alternate EVT (EVTO [preferred], EVTI1, EVTO1) signal
Tool Vendor Support
Current Nexus Aurora Tool Support

The new high-speed Nexus Aurora physical interface is supported by several of the most popular tool vendors.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Lauterbach Tools</th>
<th>Green Hills Software</th>
<th>P1s Development Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nexus Aurora trace</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Trace to memory supported</td>
<td>✓</td>
<td>—</td>
<td>✓</td>
</tr>
<tr>
<td>SPU support</td>
<td>—</td>
<td>—</td>
<td>✓</td>
</tr>
</tbody>
</table>
Summary
FTF Resources

There are many sessions at FTF that may be useful to attendees, some of these are listed below.

<table>
<thead>
<tr>
<th>Session ID</th>
<th>Title</th>
<th>When</th>
</tr>
</thead>
<tbody>
<tr>
<td>FTF-AUT-F0240</td>
<td>Unleash the Power of Qorivva MPC57xx 32-bit MCUs Using the Inter-Processor Communication (IPC) Interface</td>
<td>Thursday 3:00 PM</td>
</tr>
<tr>
<td>FTF-AUT-F0338</td>
<td>Hands-On Workshop: Performance Optimization Hints and Tips for Power Architecture® (Reserved Seat Required)</td>
<td>Tuesday 1:00 PM</td>
</tr>
<tr>
<td>FTF-AUT-F0345</td>
<td>MPC57xx e200zx Core Differences</td>
<td>Wednesday 11:30 AM</td>
</tr>
<tr>
<td>FTF-AUT-F0239</td>
<td>Advanced Debug Features of the Qorivva MPC57xx 32-bit MCUs</td>
<td>Thursday 4:15 PM</td>
</tr>
</tbody>
</table>
Web Resources

- Application Notes
  - AN4591: Lauterbach MPC57xx Nexus Trace tools
  - AN4566: MPC5746M Hardware Design Guide
  - AN4802: Qorivva MPC57xx e200zx Core Differences
- Evaluation Board Documentation
  - MPC5744PEEVB257UG - Qorivva MPC5744P Evaluation Board 257BGA Expansion User’s Guide
  - MPC5746M EVB User’s Guide
- Device Reference Manuals
  - MPC5744P Microcontroller Reference Manual
Summary

Many of the devices in the MPC57xx family of automotive microcontrollers support a new high-speed Nexus Aurora trace interface that allows advanced program and data trace
Power Architecture Tool Vendor Support

Debugger Support

Compilers

Calibration
MPC57xxM Trace Block Diagram

- 1.25 GHz clock from external debug tool to MCU
- 1.25 Gbits/sec Aurora trace lanes from MCU to debug tool
Nexus Aurora Router

• Based Nexus interface used on the P4080 with additional capability
  - JTAG access to NAR registers
• Allows trace to physical port (Aurora) or to on-chip memory
  - 16K trace memory built into the Flash on production device
  - Up to 2M trace memory in buddy die
    • Buddy die SRAM can be split between trace and calibration, 4 separate blocks
    - On the PD NAR, the output queue can trace either 100% to flash trace RAM or 100% to BD NAR
    - BD NAR can split the output queue to 50/50 between the trace RAM and the Aurora output
Nexus Aurora Link (NAL)

- Reused from P4080
- Formats the information from the NAR and stripes it into multiple lanes for the NAP
- Supports 2 to 8 lanes of Aurora data
- Encoding is 8B/10B encoding
- CRC not available on NPC version of this module
Nexus Aurora Physical Interface

- Low Voltage Differential Signaling (LVDS)
- Xilinx Aurora Protocol V2.x Compliant
- Configurable up to 8 lanes, currently only a maximum of 4 are planned
- Maximum data rate is 3 Gb/s
- Clock provided by tool (625 MHz to 1.25 GHz maximum)
Aurora Training Sequence Capture

- Training sequence
  - SYNC and polarity
  - Sync and polarity acknowledge
  - Bonding (on idle symbols)
  - Verification
- Start of channel (protocol data units)
Aurora Training Sequence

- Current implementation for automotive is 2 or 4 lanes
- Training sequence transmitted to establish initial link
- When Nexus data is not available, the IDLE sequence is transmitted
- Nexus data can be framed with a CRC added
- CRC seed is 0 at the beginning of each frame