RECENT STUDY IN HIGH SPEED SERIAL LINKS AND PROTOCOLS FOR MPAR

Hernan A. Suarez

Department of Electrical and Computer Engineering
Radar Innovations Laboratory

UNIVERSITY OF OKLAHOMA
HIGH SPEED SERIAL LINKS

Advantages over parallel links:

- More flexible, thinner cabling

- Topologies that promise to scale to the needs of the end user

- Reduced system costs because of fewer PCB traces, and lower pin/wire count.

- Predictable and reliable signaling schemes

- Exceptional bandwidth per pin.
• An embedded system is a computing system (other than a general-purpose computer) with the following general characteristics
  • Single-functioned
    • Typically, is designed to perform predefined function
  • Tightly constrained
    • Tuned for low cost
    • Single-to-fewer components based
    • Performs functions fast enough
    • Consumes minimum power
  • Reactive and real-time
    • Must continually monitor the desired environment and react to changes
  • Hardware and software co-existence
TRADITIONAL EMBEDDED SYSTEM

- Power Supply
- Ethernet MAC
- Audio Codec
- CLK
- Address Decode Unit
- Memory Controller
- CPU (uP / DSP)
- Co-Proc.
- UART
- custom IF-logic
- Display Controller
- GP I/O
- Interrupt Controller
- Timer
- SDRAM
- SDRAM
- SDRAM
NEXT STEP

- CPU (uP / DSP)
- Co-Proc.
- Memory Controller
- Address Decode Unit
- UART
- custom IF-logic
- L C
- Display Controller
- Power Supply
- Timer
- Interrupt Controller
- SDRAM
- SDRAM
- SDRAM
- SRAM
- SRAM
- SRAM
- GP I/O
- Ethernet MAC
- Audio Codec
- CLK
- FPGA
CONFIGURABLE SYSTEM ON A CHIP (CSoC)
HOW TO BUILD SUCH A SYSTEM?
XILINX PLATFORM STUDIO

Standard Embedded SW Development Flow
- C Code
- Compiler/Linker
- (Simulator)
- Object Code

CPU code in off-chip memory
Download to Board & FPGA
Debugger

CPU code in on-chip memory

Standard FPGA HW Development Flow
- VHDL/Verilog
- Synthesizer
- Simulator
- Place & Route

Download to FPGA

Manipulate
VHDL/Verilog
• **PowerPC 405 (hard core)**
  - 32 bit embedded PowerPC RISC architecture
  - Up to 450 MHz
  - 2x16 kB instruction and data caches
  - Memory management unit (MMU)
  - Hardware multiply and divide
  - Coprocessor interface (APU)
  - Embedded in Virtex-II Pro and Virtex-4
  - PLB and OCM bus interfaces

• **MicroBlaze (soft core)**
  - 32 bit RISC architecture
  - 2 64 kB instruction and data caches
  - Barrel Shifter
  - Hardware multiply and divide
  - OPB and LMB bus interfaces
PowerPC-BASED EMBEDDED SYSTEM

- PowerPC 405 Core
- Dedicated Hard IP: DSOCM BRAM, RocketIO
- Flexible Soft IP: ISOCM BRAM
- DCR Bus
- UART
- GPIO
- On-Chip Peripheral Bus: PLB, Arbiter
- Hi-Speed Peripheral
- e.g. Memory Controller
- GB E-Net
- Bus Bridge
- On-Chip Peripheral: UART, GPIO, DDR SDRAM, SDRAM

Full system customization to meet performance, functionality, and cost goals
INTEGRATION IN SYSTEM DESIGN

Programmable Systems user in a new era of system design integration possibilities
VIRTEX-II PRO

Three high current power supplies with continuous monitoring.

Platform Flash for storing FPGA configurations.

USB2 port for FPGA configurations.

Compact flash card port for FPGA config and removable storage.

DDR SDRAM DIMM slot holds up to 2GBbytes.

Virtex-II Pro XC2VP30 FPGA.

SystemACE chip for Compact Flash I/O.

SMA connectors for Gigabit serial I/O.

SATA connectors for Gigabit serial I/O.

10/100 Ethernet MAC/PHY.

Stereo audio via AC97 codec.

Power connector and switch.

XSGA Video Port.

High-speed expansion connector compatible with Digilent boards.

Buttons, switches, and LEDs.

Low-speed expansion connector compatible with Digilent boards.

Additional I/O via four 60-pin headers.

RS-232 serial port.

PS/2 mouse and keyboard port.
**XUPV2P FEATURES**

- Virtex-2 Pro XC2VP30 FPGA with:
  - 30,816 Logic Cells
  - 136 18-bit multipliers,
  - 2,448Kb of block RAM,
  - 2 PowerPC Processors
- DDR SDRAM DIMM that can accept up to 2Gbytes of RAM
- 10/100 Ethernet port
- USB2 port
- Compact Flash card slot
- XSGA Video port
- Audio Codec
- SATA, and PS/2, RS-232 ports
- High and Low Speed expansion connectors,
  - large collection of available expansion boards
TOPOLOGIES

Ring topologies via two 1x links

1x connection to switch

1x connection to switch with connections to FPGA and ASIC

Mesh with four 1x links

Five DSPs completely connected
SERIAL NETWORK

RapidIO Serial Switch

System Processing

Storage Device

RapidIO

Aurora

Serial ATA
AURORA PROTOCOL

DEFINE:
- Physical layer interface: electrical level and symbol coding
- Initialization and error handling
- Data stripping
- Link layer (PDU)
- Flow control

DOES NOT DEFINE:
- Error detection and recovery: does not define a mechanism for detecting error in users PDUs.
- Data switching: does not define an address scheme.
SERIAL CONNECTIVITY WITH AURORA IP
(Types of Data)

- Clock compensation sequences
- Initialization sequences
- Native flow control PDU
- User flow control PDU
- Channel PDU
- Idle sequences
User Data Transmission

Steps:

- Padding: all user PDUs that contain an odd number of octets must be padded with single octet.

- Link layer encapsulation: the user PDU is encapsulated with control symbol sequences. (SCP, and ECP mark channel)

- 8B/10B Encoding: link layer frame is encoded using 8B/10B (or 10B/64B)

- Serialization: after the complete channel PDU has been assembled

* When user streaming is used the second step is omitted. Idle sequences must be transmitted identically on all lines, it is not usually required when the data width is multiple of lines used. Idle sequences are inserted when no data is available.
SERIAL CONNECTIVITY WITH AURORA IP

1. **User PDU**
   - **N Octets**
   - **Padding**

2. **Link-Layer Encapsulation**
   - **Two Octets**
   - **N Octets**
   - **One Octet Two Octets**
   - **SCP**
   - **User PDU**
   - **PAD**
   - **ECP**

3. **8B/10B Encoding**
   - **Two Symbols**
   - **N Symbols**
   - **One Symbol Two Symbols**
   - **SCP**
   - **User PDU**
   - **PAD**
   - **ECP**
SERIAL CONNECTIVITY AURORA IP

Configurable parameters:

- Streaming / framing interface.
- Simplex / full duplex data flow
- Single / multiple MGTs
- Reference clock value
- Line rate
- MGT location based on number of MGTs selected
- Reference source clock
FULL DUPLEX MODE

Channel Partner A

User Flow Control Messages
User PDUs

Rx Native Flow Control State Machine

Native Flow Control
Idle Sequences

Tx Native Flow Control State Machine

Idle Sequences
Native Flow Control

MUX

DEMUX

Channel Partner B

User Flow Control Messages
User PDUs

User Rx FIFO

FIFO Status

Tx Native Flow Control State Machine

 Idle Sequences
Native Flow Control

MUX

DEMUX

Aurora Channel

Native Flow Control
Idle Sequences

User Flow Control Messages
User PDUs

User Flow Control Messages
User PDUs
SIMPLEX MODE
COMMUNICATION BETWEEN TWO XUV2P

PowerPC

OPB

Read FIFO
RX FIFO
Write FIFO
TX FIFO

Aurora Core
Aurora Core

Read FIFO
RX FIFO
Write FIFO
TX FIFO

OPB

PowerPC
Your peripheral will be connected to the DPB bus through the DPB IP interface (PIF) module. Besides standard functions like address decoding, this module also offers other commonly used services. Using these services may significantly simplify the implementation of your peripheral.

- **Basic slave service and support**
  - Common and typically required by most peripherals for operations like logic control, status report, and etc.
  - S/W reset and MIR
  - User logic interrupt support
  - User logic S/W register support

- **Master service and support**
  - Typically required by complex peripherals like Ethernet and PCI for command data transfers between regions.
  - DMA
    - Simple mode
    - Packet mode Scatter Gather
  - User logic master support

- **Advance slave service and support**
  - Typically required by peripherals that need data buffering or multiple memory/address spaces access.
  - Burst transaction support
  - FIFO
  - User logic address range support
The IPIF can be set up such that it contains a FIFO. Your peripheral can use this FIFO to interact with the processor.

- Include Read FIFO
- Use packet mode
- Use vacancy calculation

Data width (bit) of Read FIFO: 32
Number of Read FIFO entries: 512

- Include Write FIFO
- Use packet mode
- Use vacancy calculation

Data width (bit) of Write FIFO: 32
Number of Write FIFO entries: 512
SERIAL CONNECTIVITY WITH AURORA IP
R232 INTERFACE
R232 INTERFACE

DATA RECEIVED

VISA resource name: CCM1
baud rate: 9600
data bits: 8
parity: None
stop bits: 1.0
timeout (ms): 10000
termination char (0xA = "\n" = LF): 4
bytes to read: 2

Input buffer size: 4096
End write with termination character?
End read on termination character?

0x0032 0x0031 0x0030 0x002F 0x002E 0x002D 0x002C 0x002B 0x002A 0x0029 0x0028 0x0027 0x0026 0x0025 0x0024 0x0023 0x0022 0x0021 0x0020 0x001F 0x001E 0x001D 0x001C 0x001B 0x001A 0x0019 0x0018 0x0017 0x0016 0x0015 0x0014 0x0013 0x0012 0x0011 0x0010 0x000F 0x000E 0x000D 0x000C 0x000B 0x000A 0x0009 0x0008 0x0007 0x0006 0x0005 0x0004 0x0003 0x0002 0x0001
0x0032 0x0031 0x0030 0x002F 0x002E 0x002D 0x002C 0x002B 0x002A 0x0029 0x0028 0x0027 0x0026 0x0025 0x0024 0x0023 0x0022 0x0021 0x0020 0x001F 0x001E 0x001D 0x001C 0x001B 0x001A 0x0019 0x0018 0x0017 0x0016 0x0015 0x0014 0x0013 0x0012 0x0011 0x0010 0x000F 0x000E 0x000D 0x000C 0x000B 0x000A 0x0009 0x0008 0x0007 0x0006 0x0005 0x0004 0x0003 0x0002 0x0001
0x0032 0x0031 0x0030 0x002F 0x002E 0x002D 0x002C 0x002B 0x002A 0x0029 0x0028 0x0027 0x0026 0x0025 0x0024 0x0023 0x0022 0x0021 0x0020 0x001F 0x001E 0x001D 0x001C 0x001B 0x001A 0x0019 0x0018 0x0017 0x0016 0x0015 0x0014 0x0013 0x0012 0x0011 0x0010 0x000F 0x000E 0x000D 0x000C 0x000B 0x000A 0x0009 0x0008 0x0007 0x0006 0x0005 0x0004 0x0003 0x0002 0x0001
0x0032 0x0031 0x0030 0x002F 0x002E 0x002D 0x002C 0x002B 0x002A 0x0029 0x0028 0x0027 0x0026 0x0025 0x0024 0x0023 0x0022 0x0021 0x0020 0x001F 0x001E 0x001D 0x001C 0x001B 0x001A 0x0019 0x0018 0x0017 0x0016 0x0015 0x0014 0x0013 0x0012 0x0011 0x0010 0x000F 0x000E 0x000D 0x000C 0x000B 0x000A 0x0009 0x0008 0x0007 0x0006 0x0005 0x0004 0x0003 0x0002 0x0001
0x0032 0x0031 0x0030 0x002F 0x002E 0x002D 0x002C 0x002B 0x002A 0x0029 0x0028 0x0027 0x0026 0x0025 0x0024 0x0023 0x0022 0x0021 0x0020 0x001F 0x001E 0x001D 0x001C 0x001B 0x001A 0x0019 0x0018 0x0017 0x0016 0x0015 0x0014 0x0013 0x0012 0x0011 0x0010 0x000F 0x000E 0x000D 0x000C 0x000B 0x000A 0x0009 0x0008 0x0007 0x0006 0x0005 0x0004 0x0003 0x0002 0x0001
Reading data from the board_1:
WHAT'S UP?

Transmit data from Board 0 to Board_1

Writing data to the board_1:
0x0001 0x0002 0x0003 0x0004 0x0005 0x0006 0x0007 0x0008 0x0009 0x000A

Reading data from the board_1:
0x000A 0x0009 0x0008 0x0007 0x0006 0x0005 0x0004 0x0003 0x0002 0x0001

Transmit text from Board_0 to Board_1

Writing data to the Board_1:
A U R O R A 0 0 0 -

Reading data from the board_1:
WHAT'S UP?

Transmit data from Board 0 to Board_1

Writing data to the board_1:
0x0001 0x0002 0x0003 0x0004 0x0005 0x0006_
SERIAL CONNECTIVITY AURORA IP
CONCLUSION

- Serial interfaces require high-bandwidth management inside the chip

- Serial Links are more flexible, thinner cabling.

- Configuring transceivers for a particular application is challenging, as you are expected to tune more than 200 attributes.

- The Aurora IP is efficient, scalable open protocol for serial I/O.

- Aurora is designed to use minimal FPGA resources
REFERENCES

- Xilinx Platform Studio User Guide, Embedded Development Kit EDK 8.2
- Xilinx Embedded System Tools Reference Manual, Embedded Development Kit EDK 8.2i
- Xilinx Platform Specification Format Reference Manual, Embedded Development Kit EDK 8.2i
- Xilinx OS and Libraries Document Collection
- Xilinx EDK PowerPC Tutorial
- Avnet User’s Guide, Xilinx Virtex-II Pro Development Kit
THANK YOU