



Hardware Design

MicroBlaze 7.1



Objectives

After completing this module, you will be able to:

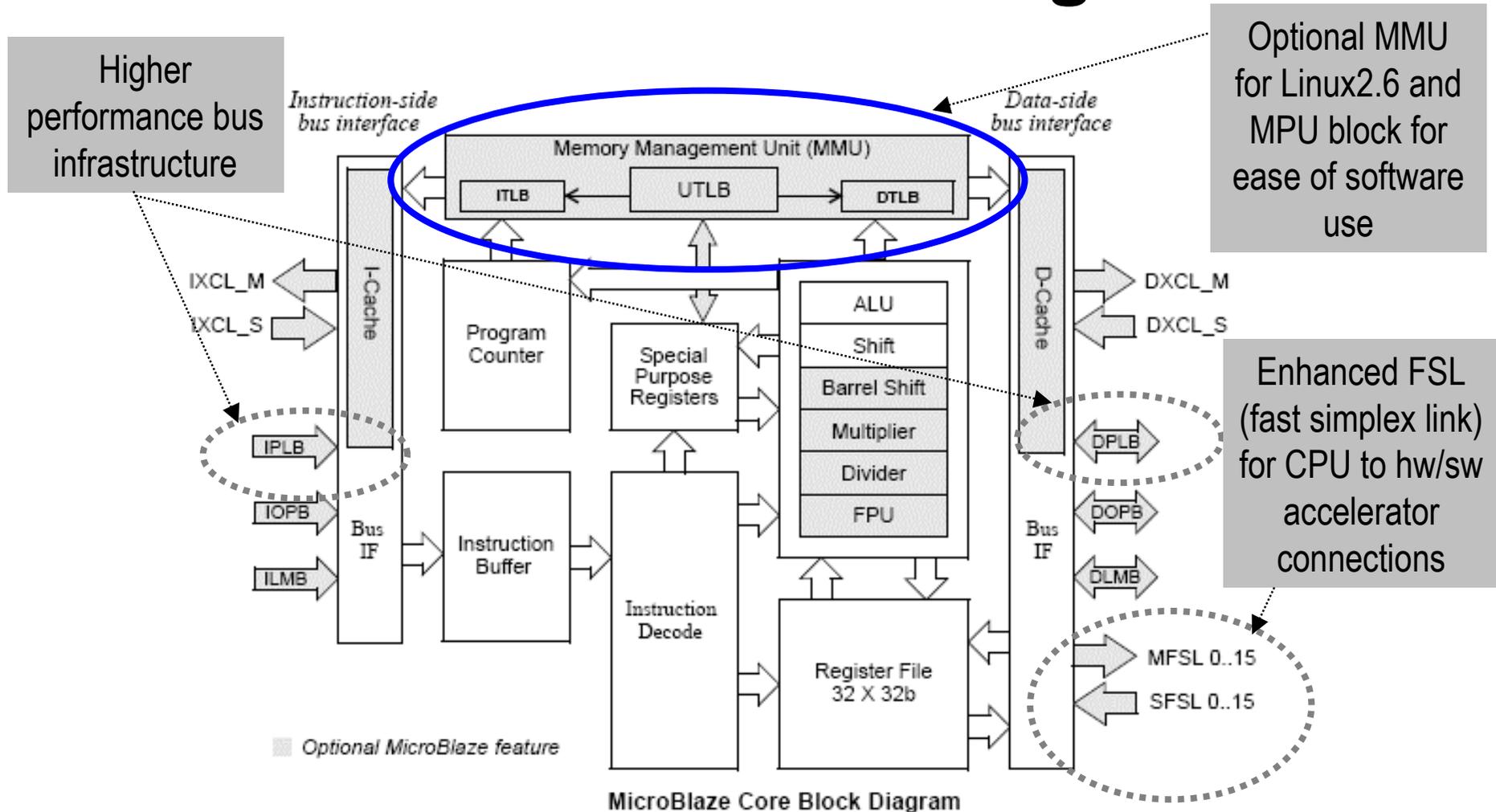
- List the MicroBlaze 7.1 Features
- List the functionality that defines an arbiter, a master, and a slave
- List the various buses available in the MicroBlaze™ processor
- List the various high performance links in the MicroBlaze processor

Outline



- **MicroBlaze Features**
- Buses 101: Arbiter, Master, Slave
- MicroBlaze System Interfaces
 - Processor Local Bus (PLB)
 - On-Chip Peripheral Bus (OPB)
 - Local Memory Bus (LMB)
 - Fast Simplex Link (FSL)
 - Xilinx Cache Link (XCL)

MicroBlaze Block Diagram



The OPB bus interface on the core is available and must be enabled through the MicroBlaze options dialogue



MicroBlaze Processor

- Scalable 32-bit Core
 - Single-Issue pipeline
 - Supports either 3-stage (resource focused) or 5-stage pipeline (performance focused)
 - Configurable Instruction and Data Caches
 - Direct mapped (1-way associative)
 - Optional Memory Mgt or Memory Protection Unit
 - Required for Linux OS (Linux 2.6 is currently supported)
 - Floating-point unit (FPU)
 - Based upon IEEE 754 format
 - Barrel Shifter
 - Hardware multiplier
 - 32x32 multiplication to generate a 64-bit result
 - Hardware Divider
 - Fast Simplex Link FIFO Channels for Easy, Direct Access to Fabric and Hardware Acceleration
 - Hardware Debug and Trace Module

Multi-processor Capability

- EDK9.1/MBv6 introduced Processor ID
- EDK9.2 introduced two fundamental MP cores
 - Mailbox: allows for message passing between 2 CPUs
 - Mutex core: allows 2 or more CPUs to synchronize
- No need for customers to build their own cores

Multi-processor Capability

Processor ID is important for multi-core processing

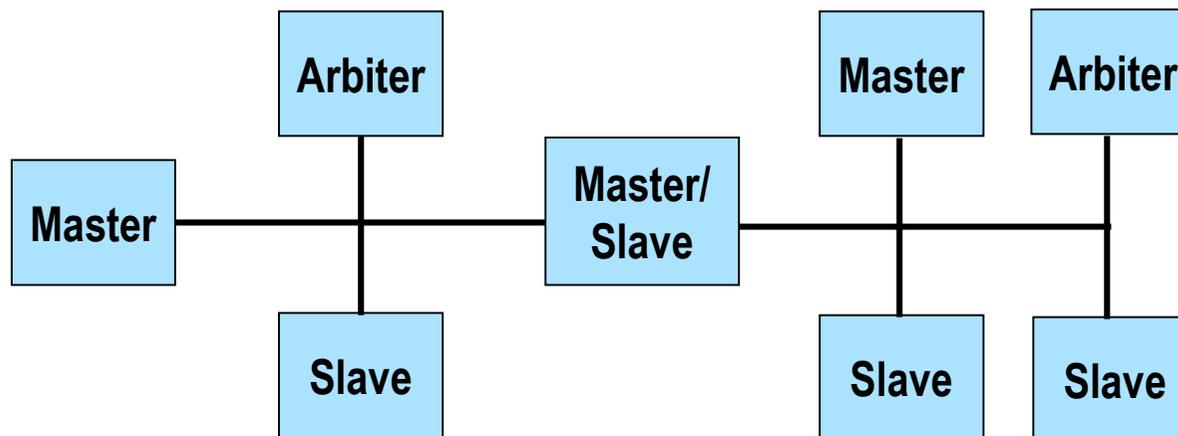
- Processor Version Register (PVR)
 - Contains: Processor ID, configuration/user/processor info (e.g. cache size etc), version number and other internal information
- PVR options:
 - None. No PVR implemented at all
 - Basic. Only first PVR register implemented
 - Allow designers to assign unique ID for a particular MicroBlaze core
 - Important for multi-core application or debug
 - Full. All PVR registers implemented
 - Allow designers access to all PVR registers for individual MicroBlaze configuration information per core

Outline

- MicroBlaze features
- • **Buses 101: Arbiter, Master, Slave**
- MicroBlaze System Interfaces
 - Processor Local Bus (PLB)
 - On-Chip Peripheral Bus (OPB)
 - Local Memory Bus (LMB)
 - Fast Simplex Link (FSL)
 - Xilinx Cache Link (XCL)

Buses 101

- A bus is a multiwire path on which related information is delivered
 - Address, data, and control buses
- Processor and peripherals communicate through buses
- Peripherals may be classified as:
 - Arbiter, master, slave, or master/slave (bridge)



Buses 101

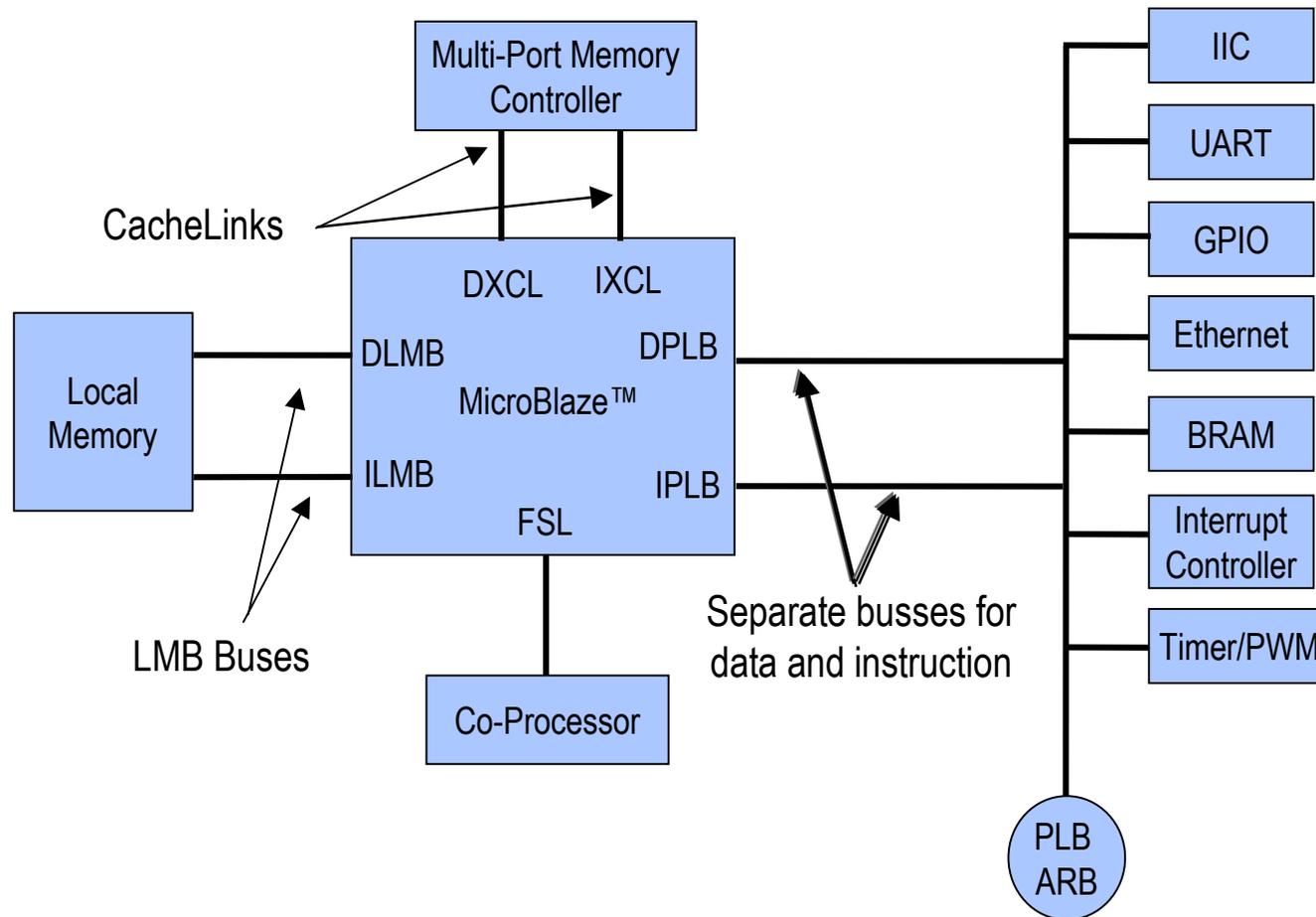
- Bus masters have the ability to initiate a bus transaction
- Bus slaves can only respond to a request
- Bus arbitration is a three-step process:
 - A device requesting to become a bus master asserts a bus request signal
 - The arbiter continuously monitors the request and outputs an individual grant signal to each master according to the master's priority scheme and the state of the other master requests at that time
 - The requesting device samples its grant signal until the master is granted access. The master then initiates a data transfer between the master and a slave when the current bus master releases the bus
- Arbitration mechanisms
 - Fixed priority, round-robin, hybrid

CoreConnect Bus Architecture

- The IBM CoreConnect bus architecture standard provides three buses for interconnecting cores, library macros, and custom logic:
 - Processor Local Bus (PLB)
 - On-Chip Peripheral Bus (OPB)
 - Device Control Register (DCR) bus
- IBM offers a no-fee, royalty-free CoreConnect bus architecture license
 - Licenses receive the PLB arbiter, OPB arbiter, and PLB/OPB bridge designs along with bus-model toolkits and bus-functional compilers for the PLB, OPB, and DCR buses
 - Required only if you create your own CoreConnect bus architecture peripheral or you are using the Bus Functional Model (BFM)

Busses 101

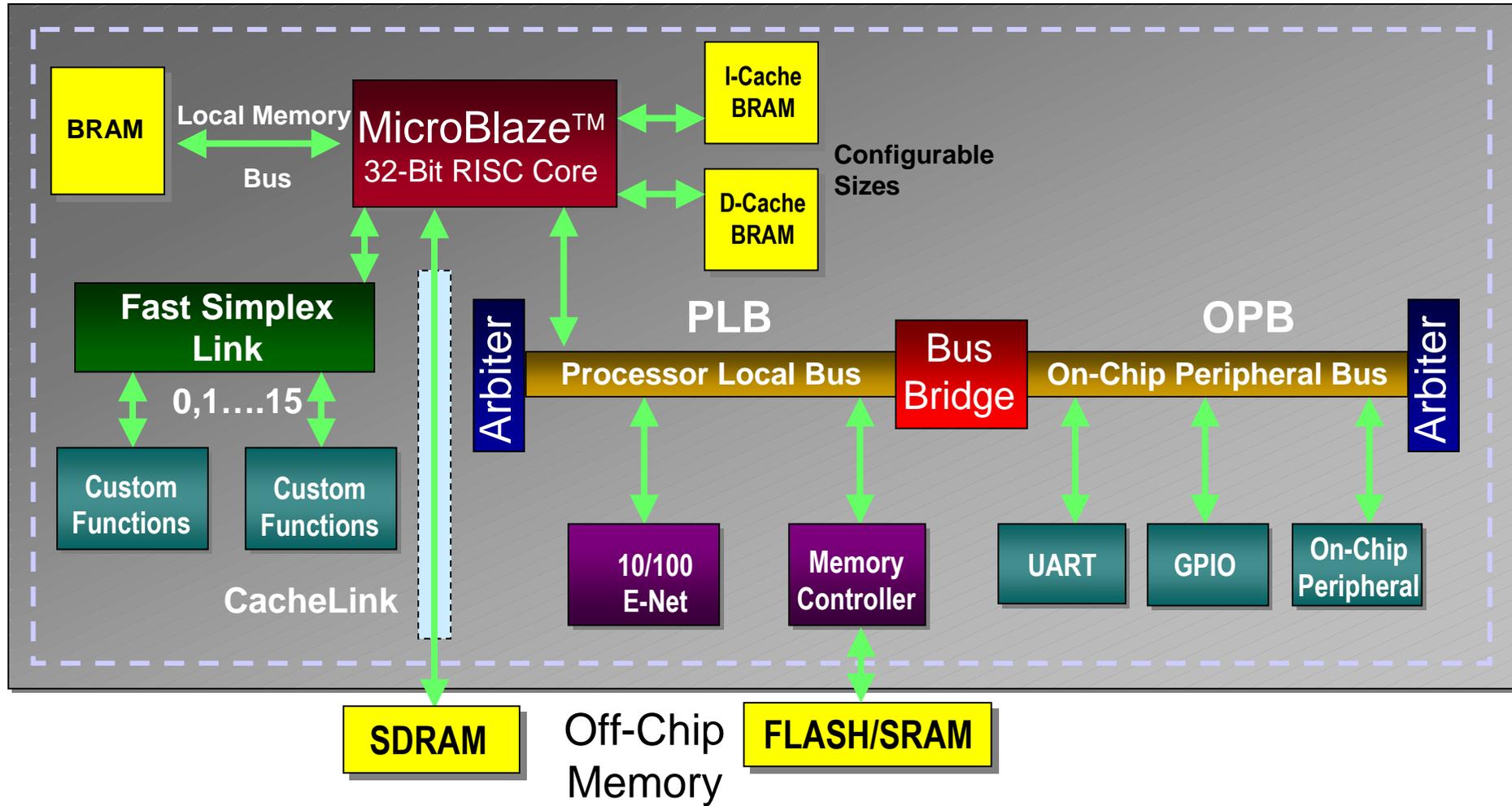
The MicroBlaze processor core is organized as a Harvard architecture



Outline

- MicroBlaze Introduction
- Buses 101: Arbiter, Master, Slave
- • **MicroBlaze System Interfaces**
 - Processor Local Bus (PLB)
 - On-Chip Peripheral Bus (OPB)
 - Local Memory Bus (LMB)
 - Fast Simplex Links (FSL)
 - Xilinx Cache Link (XCL)

MicroBlaze System

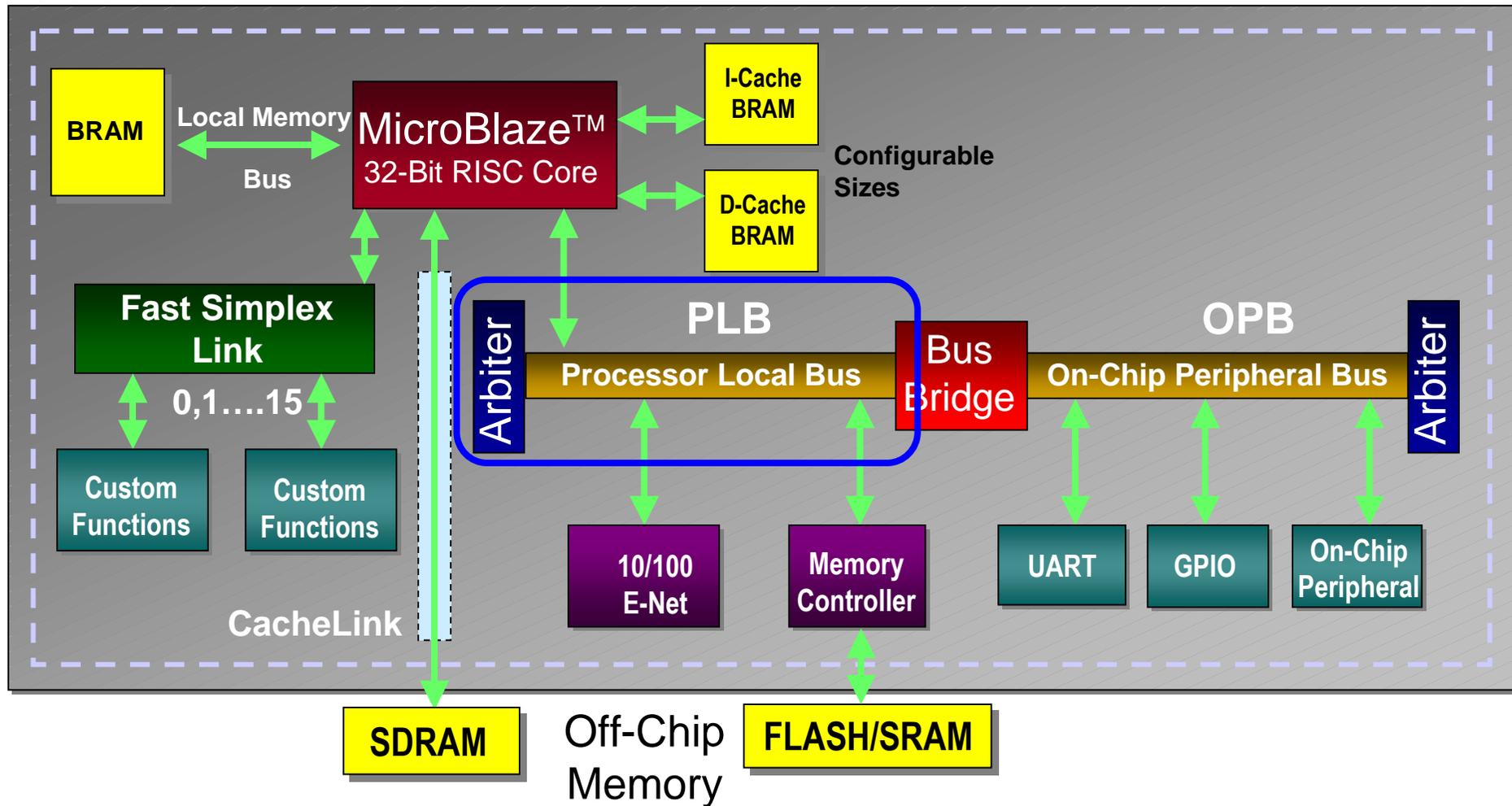


Outline

- MicroBlaze Introduction
- Buses 101: Arbiter, Master, Slave
- • **MicroBlaze System Interfaces**
 - **Processor Local Bus (PLB)**
 - On-Chip Peripheral Bus (OPB)
 - Local Memory Bus (LMB)
 - Fast Simplex Links (FSL)
 - Xilinx Cache Link (XCL)

MicroBlaze System

Processor Local Bus (v4.6 PLB)



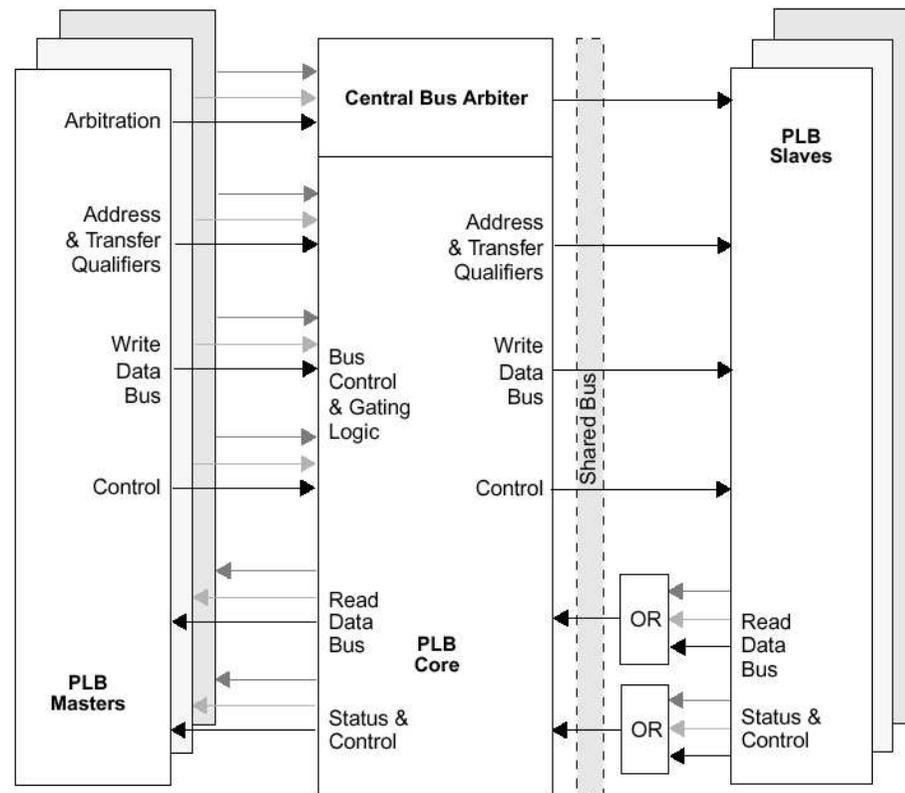
PLB Bus

- Connection infrastructure for high-bandwidth master and slave devices
- Fully synchronous to one clock
- Centralized bus arbitration—PLB arbiter
- 32-bit address
- 32, 64, or 128-bit data bus
- Selectable shared bus or point-to-point interconnect topology
 - Point-to-point optimization available for 1 master, 1 slave configuration
 - Point-to-point topology supports 0 cycle latency via arbitration removal
- Selectable address pipelining support (2-level only)
- Dynamic master request priority based arbitration
- Vectored resets and address/qualifier registers

PLB

Interconnect / Architecture

- One to 16 PLB masters, each connect all of their signals to the PLB arbiter
- The PLB arbiter multiplexes signals from masters onto a shared bus to which all the inputs of the slaves are connected
- One to n PLB slaves OR together their outputs to drive a shared bus back to the PLB arbiter
- The PLB arbiter handles bus arbitration and the movement of data and control signals between masters and slaves



PLB Bridge

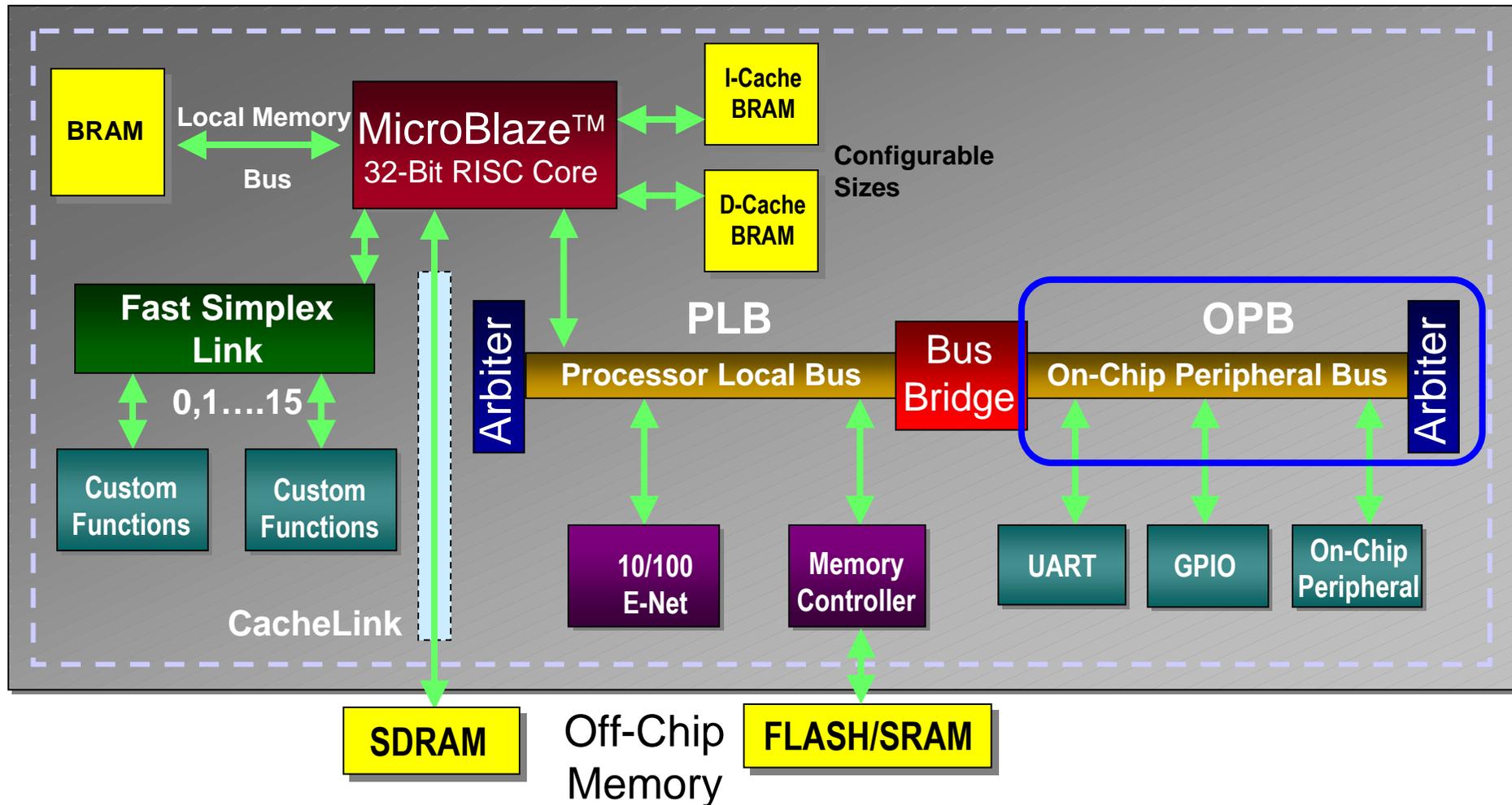
- The PLB-to-OPB bridge translates PLB transactions into OPB transactions
- This bridge functions as a slave on the PLB side and a master on the OPB side
- The bridge contains a DCR slave interface to provide access to its bus error status registers
- The bridge is necessary in systems where a PLB master device, such as a CPU, requires access to OPB peripherals

Outline

- MicroBlaze Introduction
- Buses 101: Arbiter, Master, Slave
- • **MicroBlaze System Interfaces**
 - Processor Local Bus (PLB)
 - **On-Chip Peripheral Bus (OPB)**
 - Local Memory Bus (LMB)
 - Fast Simplex Links (FSL)
 - Xilinx Cache Link (XCL)

MicroBlaze System

On-Chip Peripheral Bus (OPB)



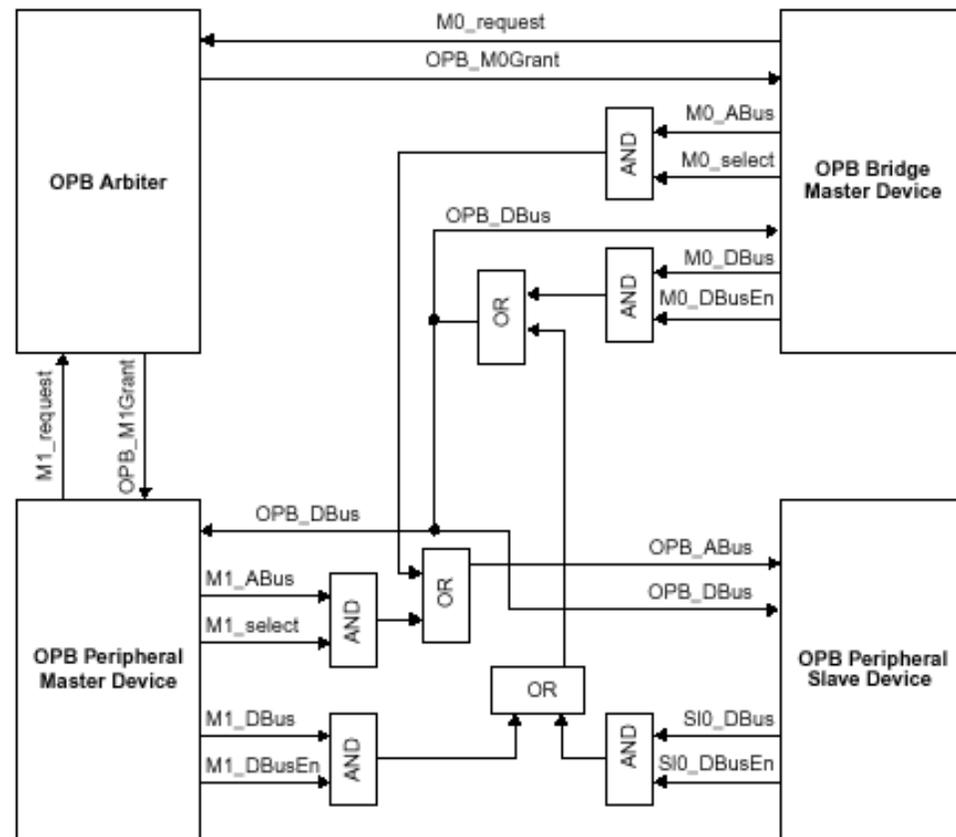
On-Chip Peripheral Bus (OPB)

- The On-Chip Processor Bus (OPB) decouples lower bandwidth devices from the PLB
- It is a less complex protocol than the PLB
 - No split transaction or address pipelining capability
- Centralized bus arbitration—OPB arbiter
- Connection infrastructure for the master and slave peripheral devices
- The OPB bus is designed to alleviate system performance bottlenecks by reducing capacitive loading on the PLB
 - Fully synchronous to one clock
 - Shared 32-bit address bus; shared 32-bit data bus
 - Supports single-cycle data transfers among the master and the slaves
 - Supports multiple masters, determined by arbitration implementation
 - The bridge function can be the master on the PLB or OPB



On-Chip Peripheral Bus (OPB)

- Supports 16 masters and an unlimited number of slaves (only limited by the expected performance)
- The OPB arbiter receives bus requests from the OPB masters and grants the bus to one of them
 - Fixed and dynamic (LRU) priorities
- Bus logic is implemented with AND-OR logic. Inactive devices drive zeros
- Read and write data buses can be separated to reduce loading on the OPB_DBus signal

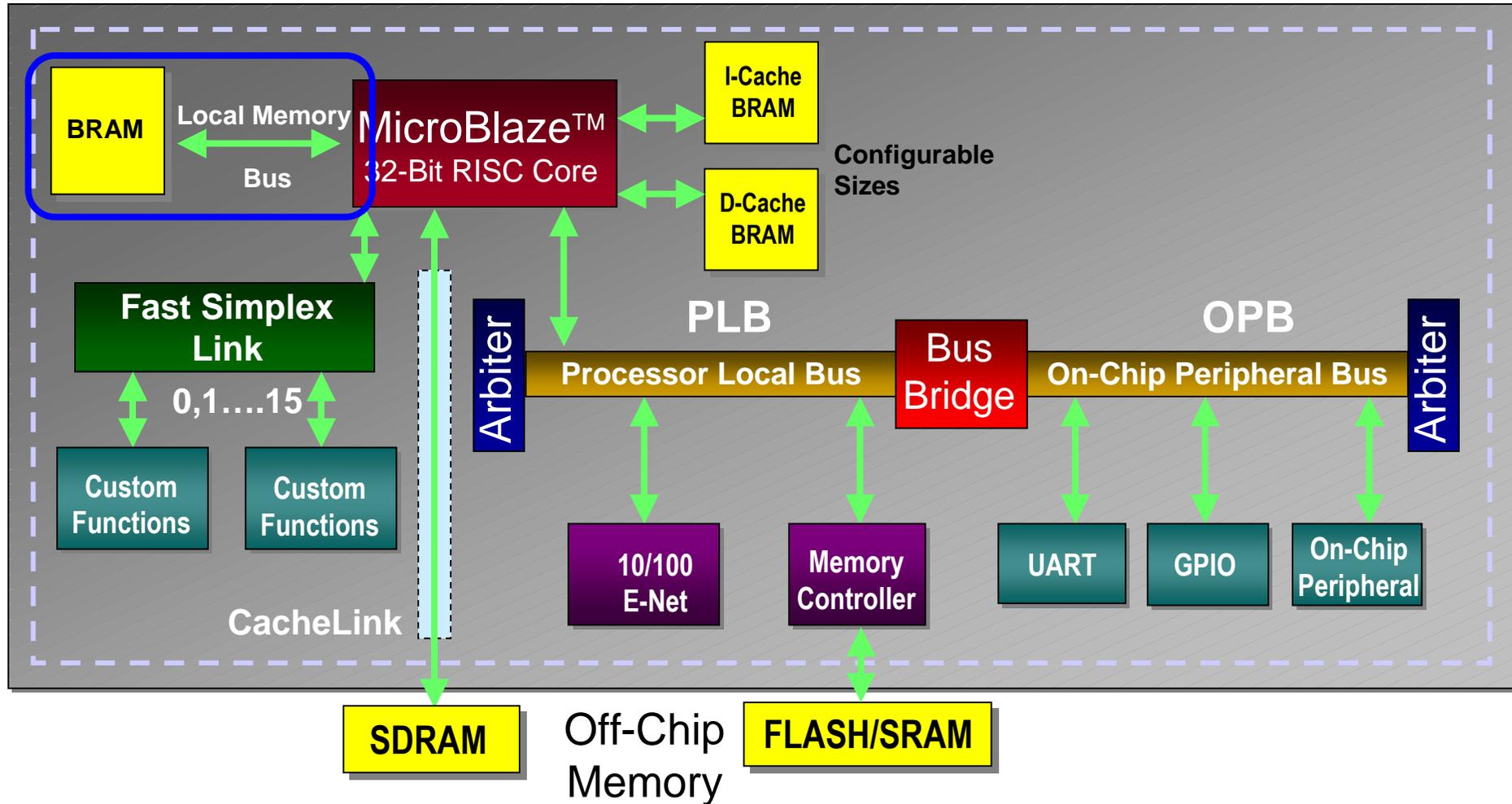


Outline

- MicroBlaze Introduction
- Buses 101: Arbiter, Master, Slave
- • **MicroBlaze System interfaces**
 - Processor Local Bus (PLB)
 - On-Chip Peripheral Bus (OPB)
 - **Local Memory Bus (LMB)**
 - Fast Simplex Links (FSL)
 - Xilinx Cache Links (XCL)

MicroBlaze System

Local Memory Bus



Local Memory Bus (LMB)

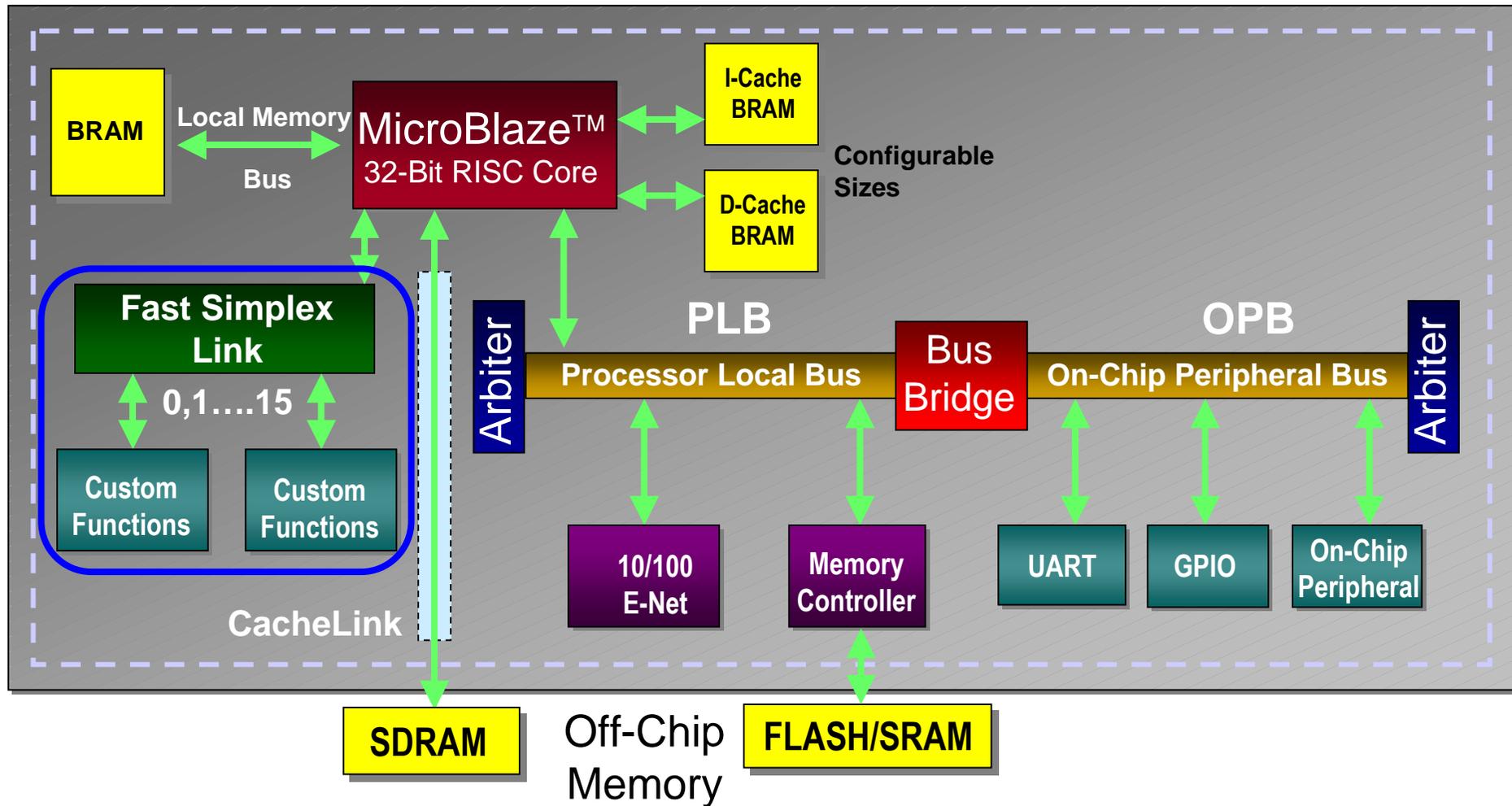
- The Local Memory Bus (LMB) provides single-cycle access to on-chip dual-port block RAM for MicroBlaze™ processors
- The LMB provides simple synchronous protocol for efficient block RAM transfers
- DLMB: Data interface, local memory bus (block RAM only)
- ILMB: Instruction interface, local memory bus (block RAM only)

Outline

- MicroBlaze Introduction
- Buses 101: Arbiter, Master, Slave
- • **MicroBlaze System Interfaces**
 - Processor Local Bus (PLB)
 - On-Chip Peripheral Bus (OPB)
 - Local Memory Bus (LMB)
 - **Fast Simplex Link (FSL)**
 - Xilinx Cache Link (XCL)

MicroBlaze System

Fast Simplex Links

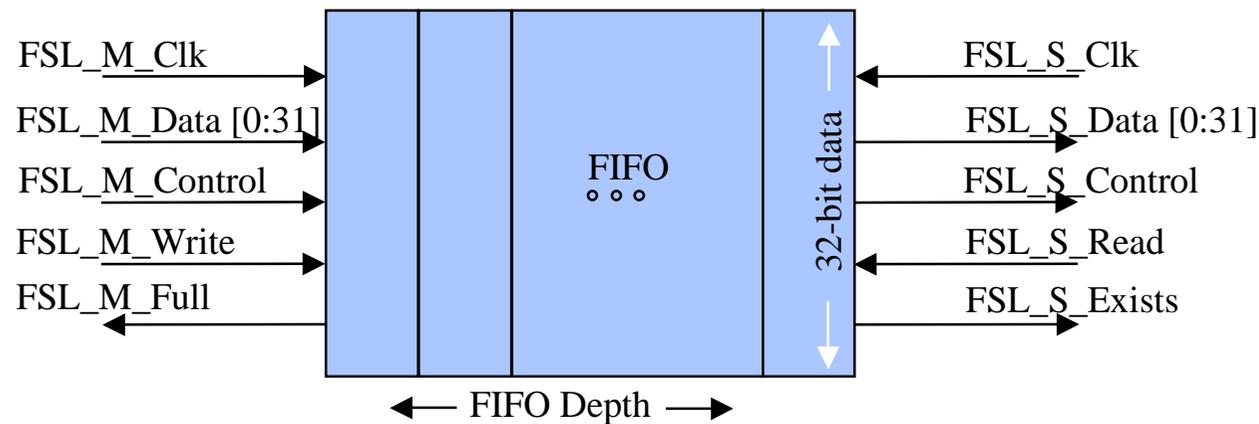


The Software Streaming Data Challenge

- Suppose you want to move data through a hardware/software processing application with following characteristics
 - Data may be of a streaming or burst nature
 - Deterministic latency between hardware and software
- *Possible solutions include*
 - Bus peripheral, maybe PLB
 - Multiple clock-cycle overhead
 - Address decode time
 - Arbitration, loss of hardware/software coherency
 - Custom microprocessor instruction access to peripheral hardware
 - May require processor to be stalled
 - Complex logic can slow overall processor speed
 - May require assembly language to access special instruction
 - Fast Simplex Links

Another Alternative: Fast Simplex Links (FSL)

- Unidirectional point-to-point FIFO-based communication
- Dedicated (unshared) and nonarbitrated architecture
- Dedicated MicroBlaze™ C and ASM instructions for easy access
- High speed, access in as little as two clocks on processor side, 600 MHz at hardware interface
- Available in Xilinx Platform Studio (XPS) as a bus interface library core from **Hardware** → **Create or Import Peripheral Wizard**



FSL Features

- 32-bit wide interface
- Configurable FIFO depths – 1 to 8193 using SRL16 or block RAM
- Synchronous or asynchronous FIFO clocking with respect to the MicroBlaze™ system clock
- Selectable use of control bit
- Simple software interface using predefined C instructions; Automatically generated C drivers
- Blocking and nonblocking software instructions for data and control (*get* and *put*)
 - Blocking FSL instruction made interruptable
 - Return from interrupt will resume the FSL instruction
- Exception from FSL can be generated
- Disable interrupt while FSL executing
- Addition of dynamic assignment of FSL channel
 - Channel number from register rather than immediate
 - GETD, PUTD instructions added

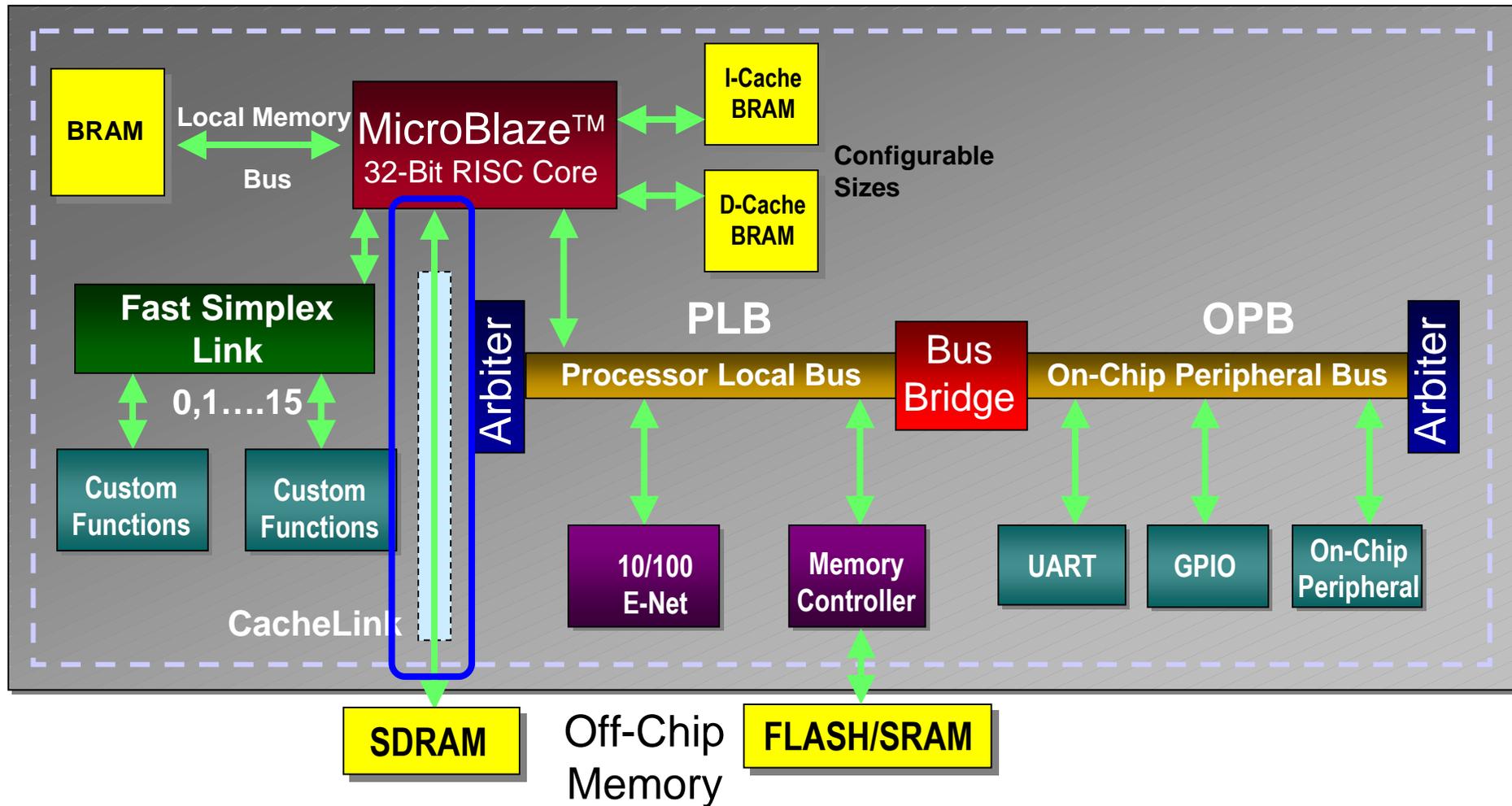


Outline

- MicroBlaze Introduction
- Buses 101: Arbiter, Master, Slave
- • **MicroBlaze System Interfaces**
 - Processor Local Bus (PLB)
 - On-Chip Peripheral Bus (OPB)
 - Local Memory Bus (LMB)
 - Fast Simplex Link (FSL)
 - **Xilinx Cache Link (XCL)**

MicroBlaze System

Xilinx Cache Links



Xilinx Cache Link

High-performance solution for memory accesses

- The MicroBlaze CacheLink interface is designed to connect directly to a memory controller with integrated FSL Buffers
 - i.e. MicroBlaze can connect directly to data ports of EDK supported multi-port memory controllers
- The CacheLink Interface is only available on MicroBlaze when caches are enabled
- The CacheLink cache controllers handle 4 or 8-word cache lines
- All individual CacheLink accesses follow the FSL FIFO based transaction protocol

Knowledge Check

- What is FSL?
- How many FSL channels are supported by MicroBlaze?
- How FSL can improve performance?

Answers

- What is FSL?
 - FSL is a dedicated simplex link with a FIFO interface
- How many FSL channels are supported by MicroBlaze?
 - 16
- How FSL can improve performance?
 - Since FSL is a dedicated link with a fixed latency
 - Bus arbitration does not exist
 - FSL is useful for hardware acceleration

Where Can I Learn More?

- Tool documentation
 - *Processor IP Reference Guide*
- Processor documentation
 - *MicroBlaze™ Processor Reference Guide*
- Support Website
 - EDK Website: www.xilinx.com/edk