« Embedded Systems »
Processeur – FPGA → EPXA1

rene.beuchat@epfl.ch
LAP/I&C/EPFL
Chargé de cours

LSN/EIG
Prof. HES

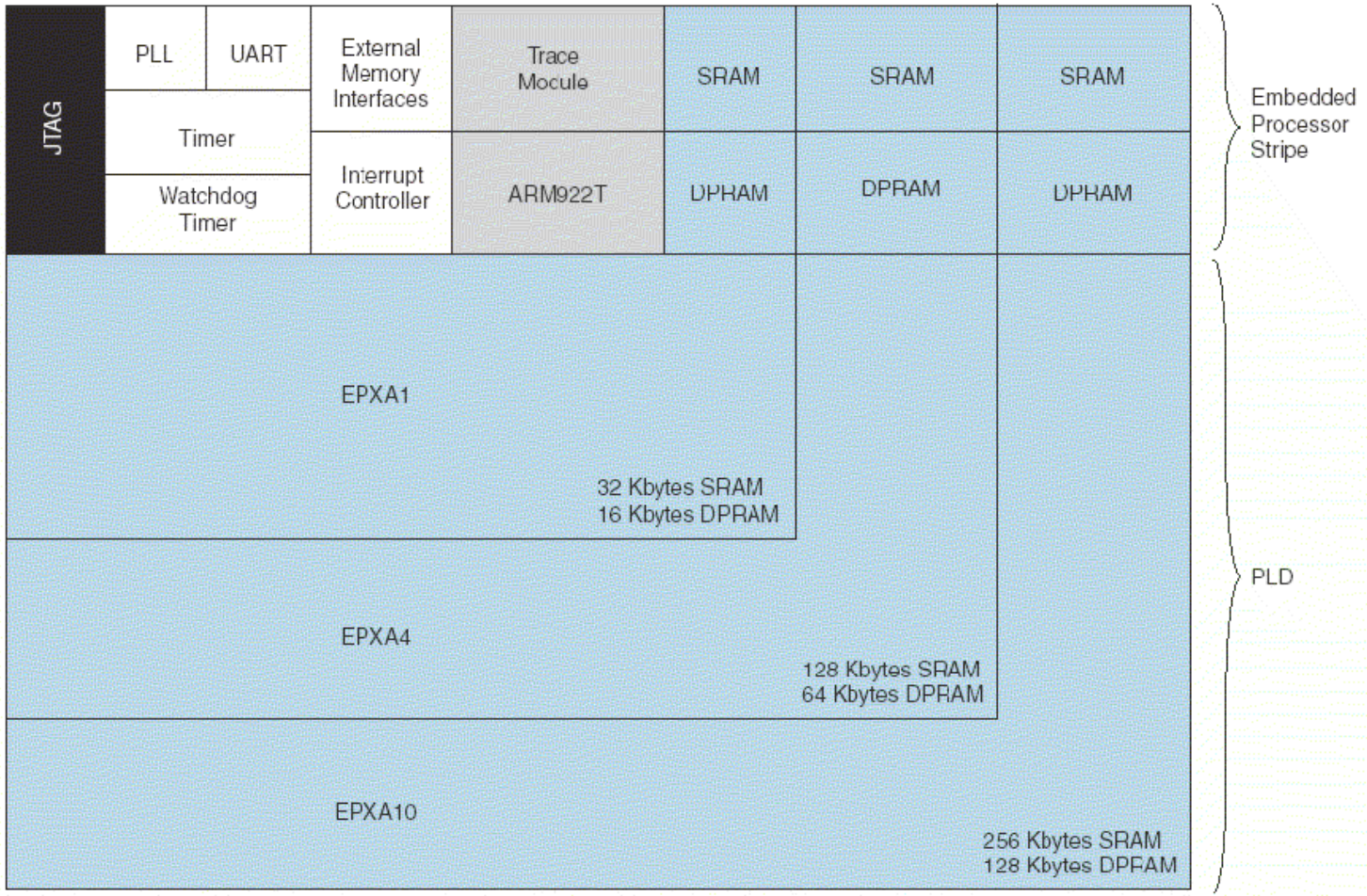
Objectives

- A specific FPGA with **hardcore** processor architecture description
- **ARM9** processor architecture
- Use of the **Amba** bus
- Bridges between bus Amba-Avalon

- Example with the Altera-EPXA FPGA

- Caution : ***old obsolete*** component, but interesting general architecture

EPXA, architecture globale



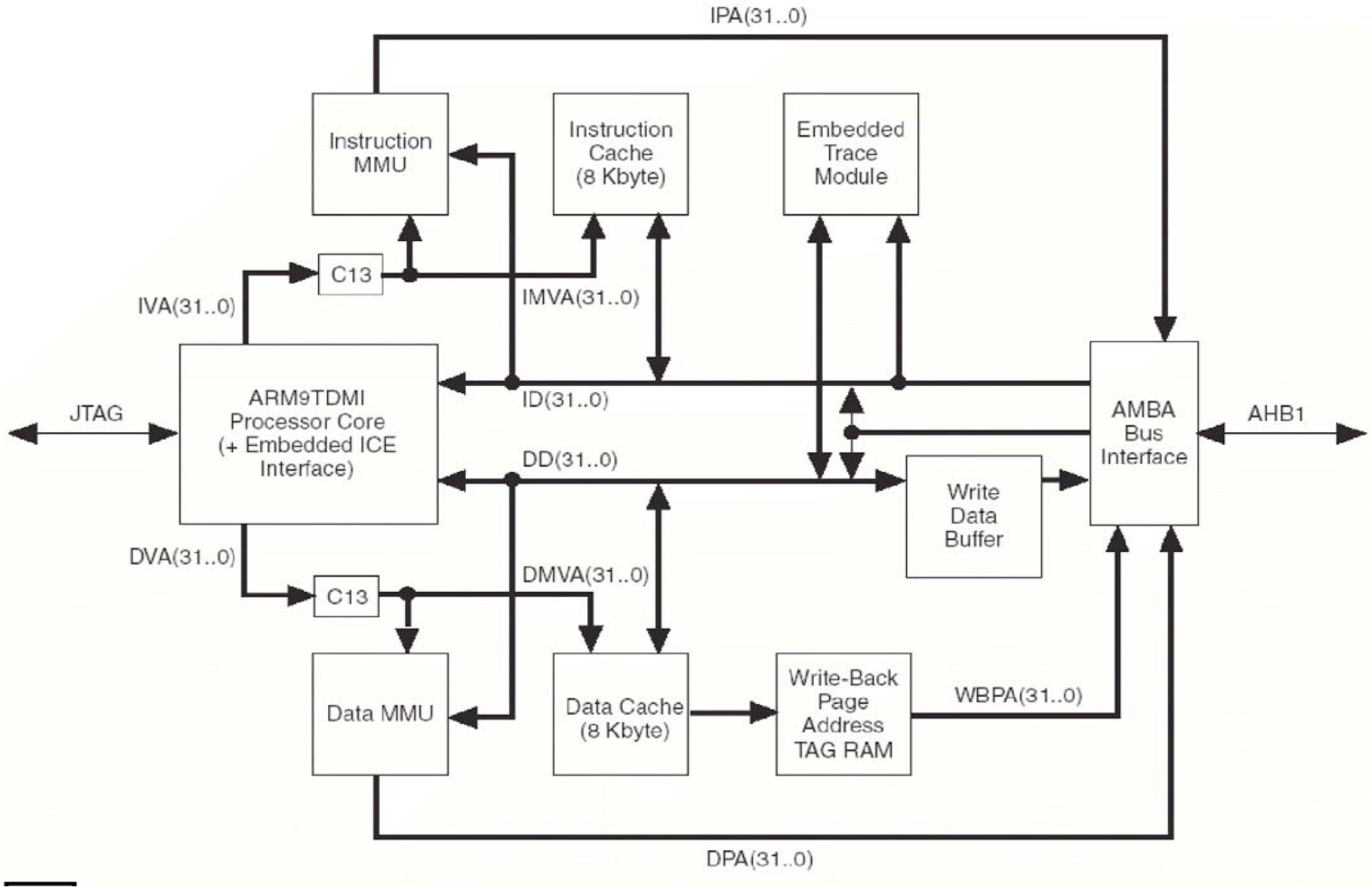
EPXA1
Interface/Memory decoding space

Elements	EPXA1	EPXA4	EPXA10
Internal registers	16kB		
Internal **SRAM**	64kB	128kB	256kB
Internal **DPSRAM**	16kB	64kB	128kB
External Bus Interface **EBI** 0..3	4 x 16kB to 32 MB		
SDRAM	16kB .. 256MB		
PLD 0..3	16kB .. 2GB		

EPXA1

SRAM	Static RAM (Random Access Memory)
DPSRAM	Dual Port SRAM
EBI	External Bus Interface
SDRAM	Synchronous Dynamic RAM
PLD	Programmable Logic Device

EPXA1, processeur ARM9

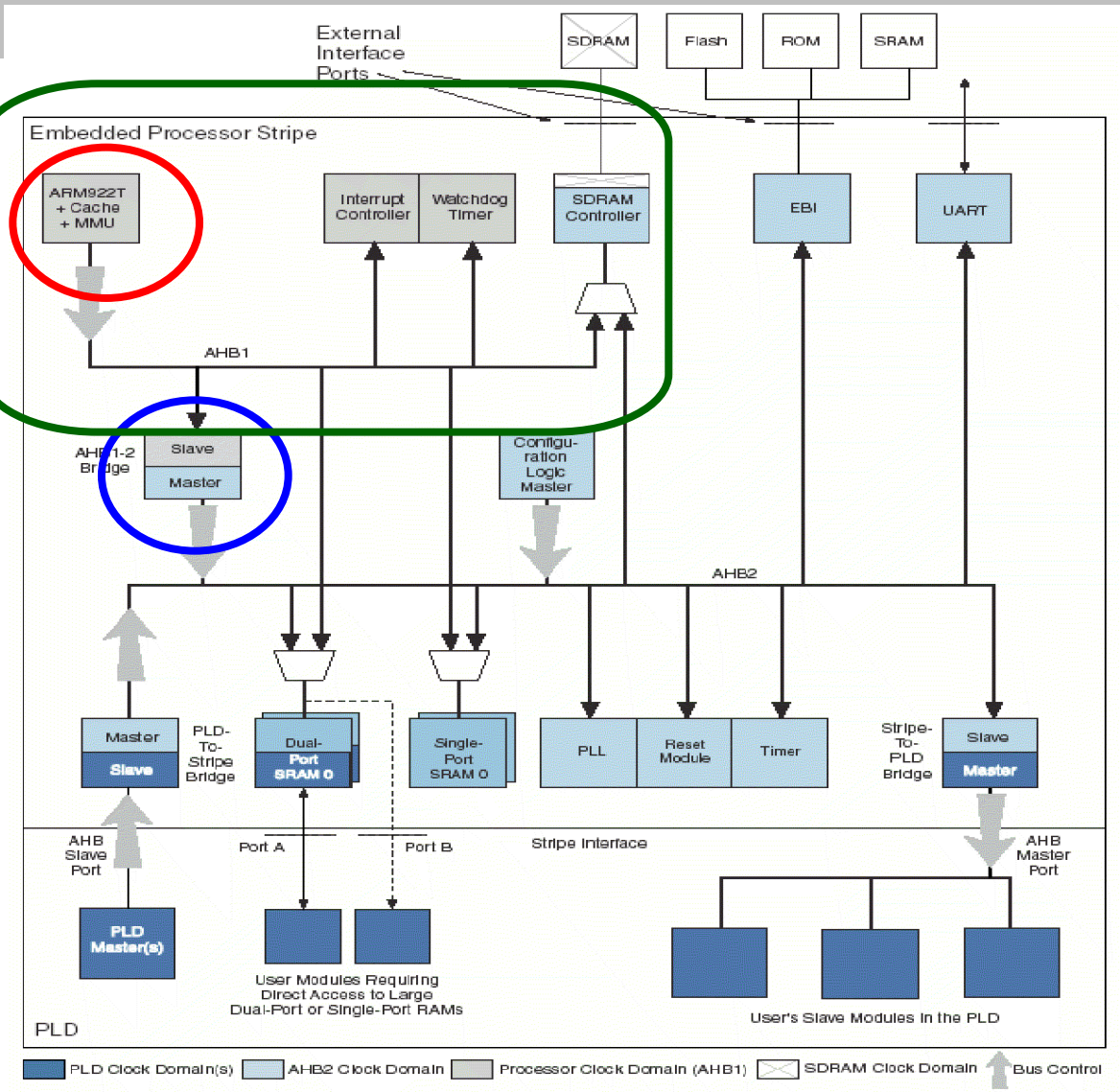


EPXA1, processeur ARM9

- Instructions cache 8 kB
- Data cache 8 kB
- Instruction MMU (Memory Management Unit)
- Data MMU
- Embedded Trace Module (ETM)
- Amba bus interface

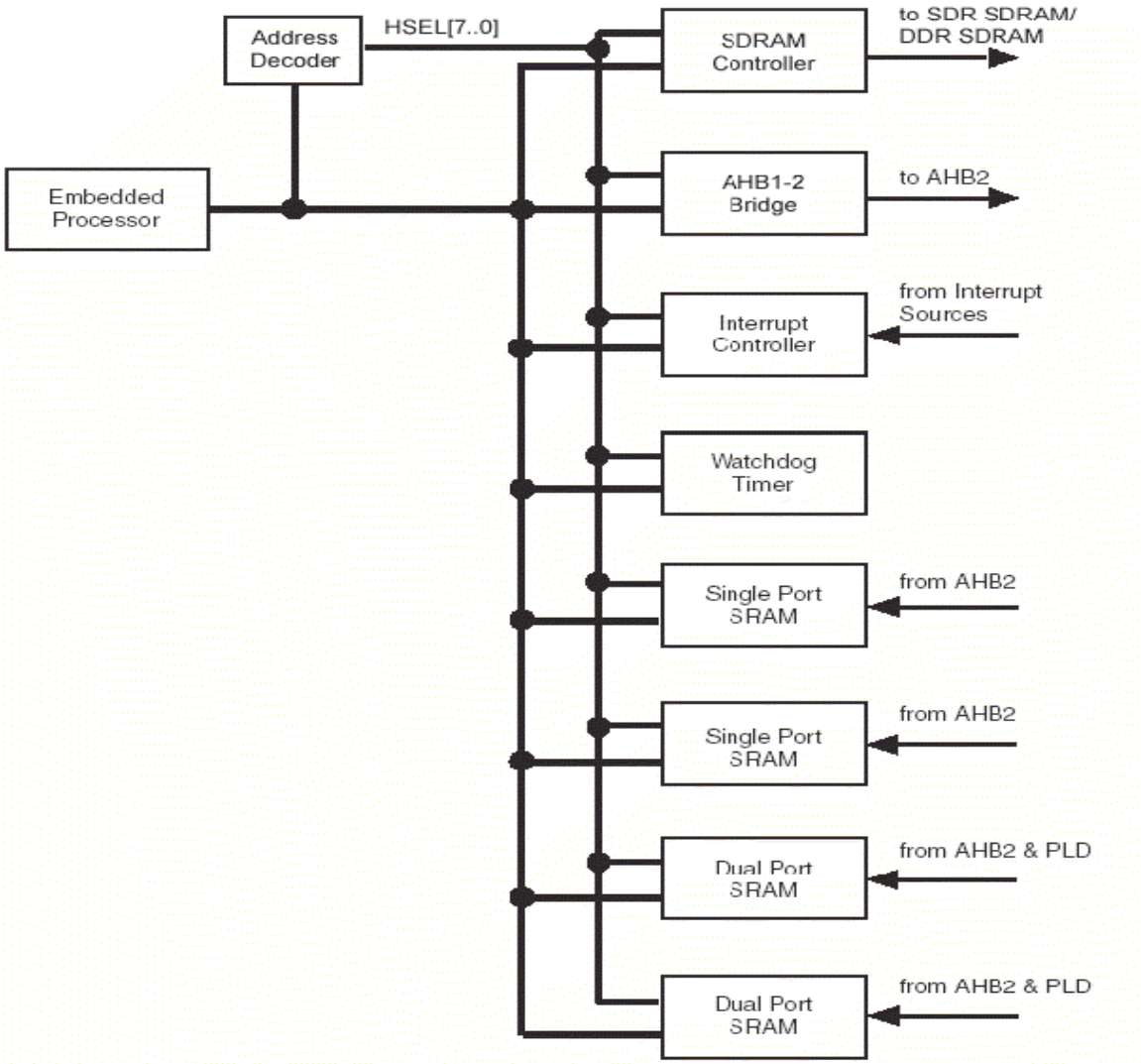
Internal / external Bus

- AHB1 :
  - ARM 922T
  - Interrupt Ctrl
  - Watch-Dog
  - SRAM/DPSRAM/
  - SDRAM Ctrl
  - Bridge → AHB2
- AHB2 :
  - Hardware stripe
  - PLL/Reset/Timer
  - UART, EBI
  - SDRAM Ctrl
  - AHB bridges ↔ PLD



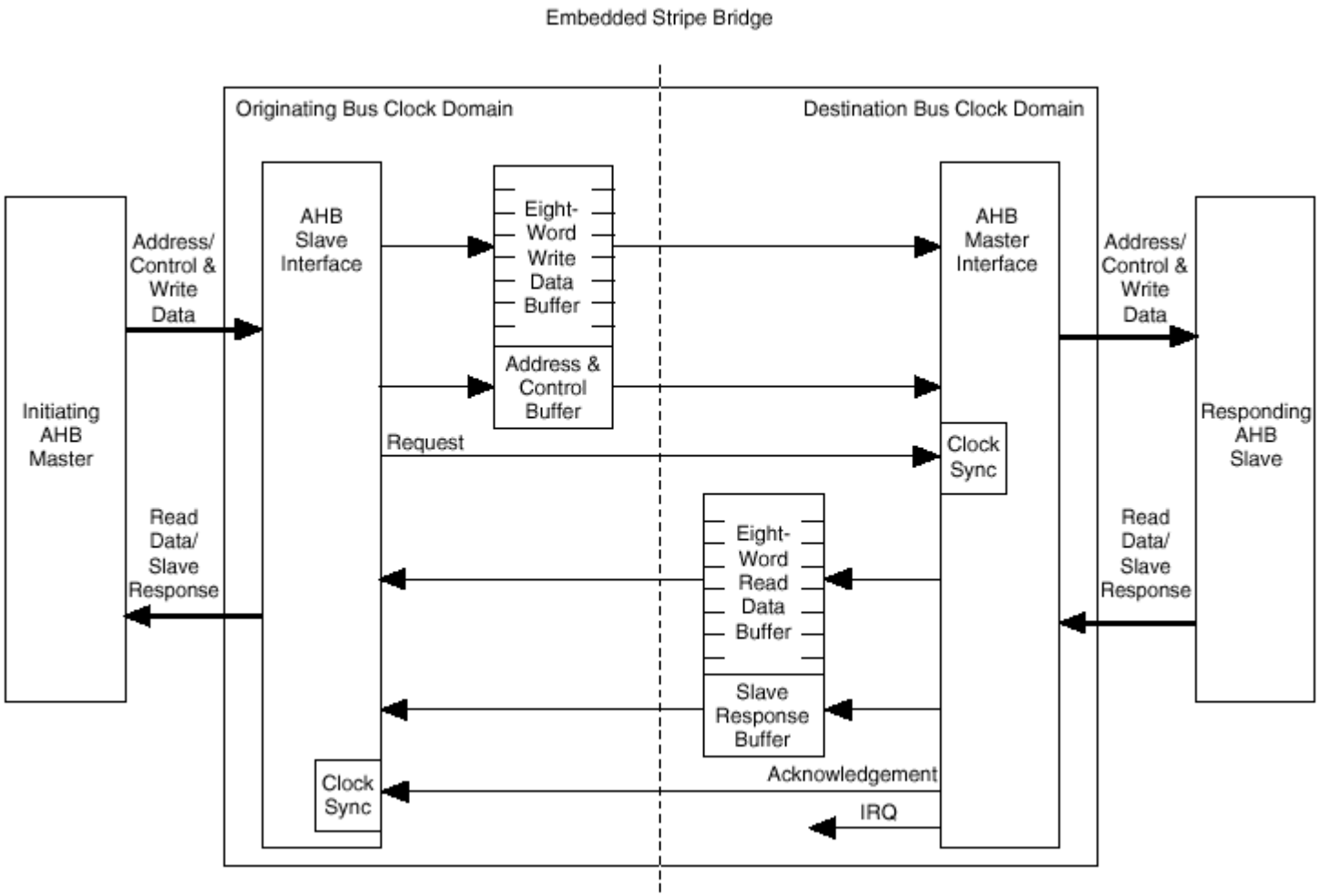
Bus AHB1

- 1 Masters :
  - ARM 922T
- 8 Slaves
  - Memories
  - Interfaces



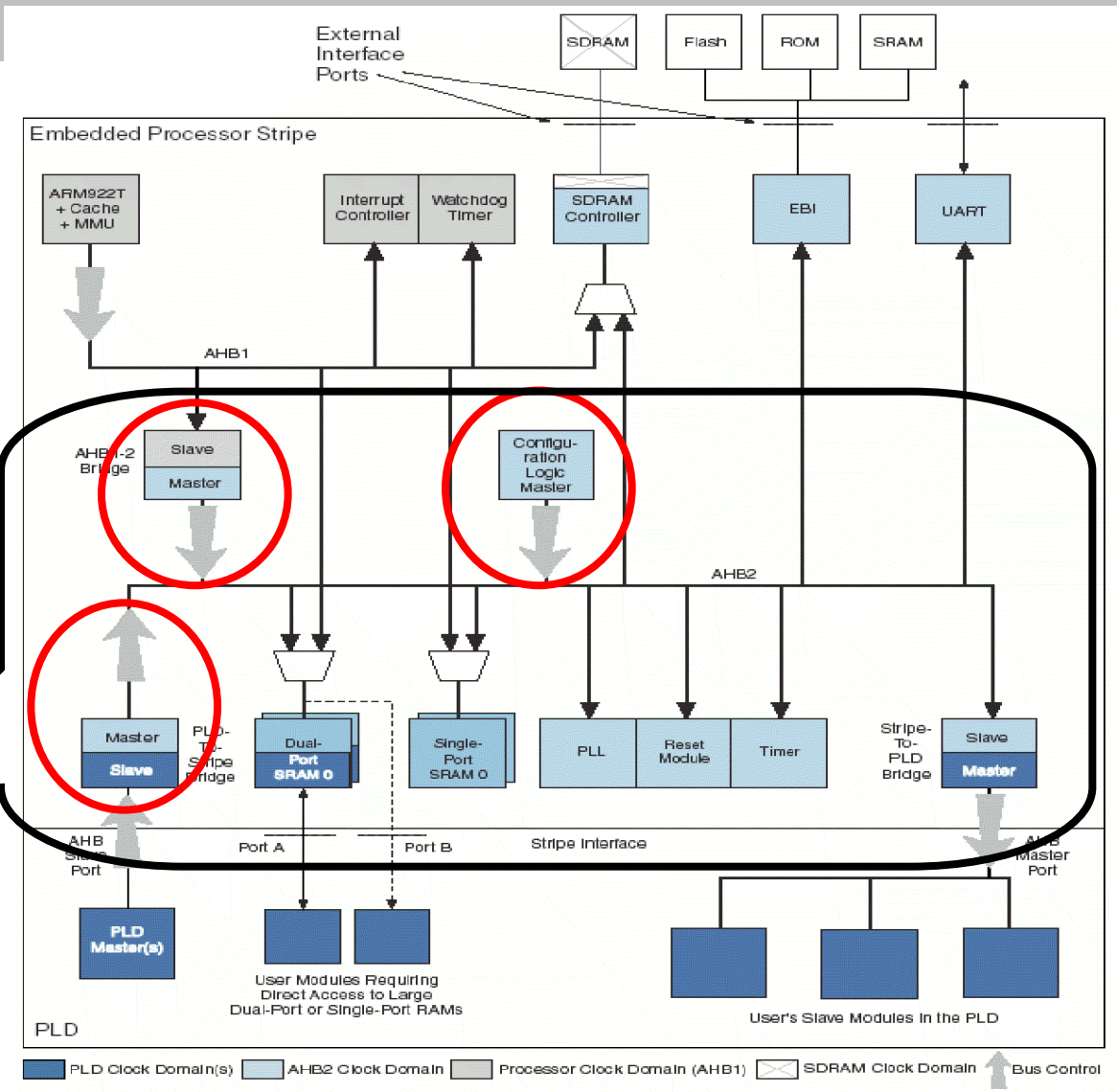
ds_exc_AHB1_bus_architecture

Stripe Bridge AHB1←→AHB2



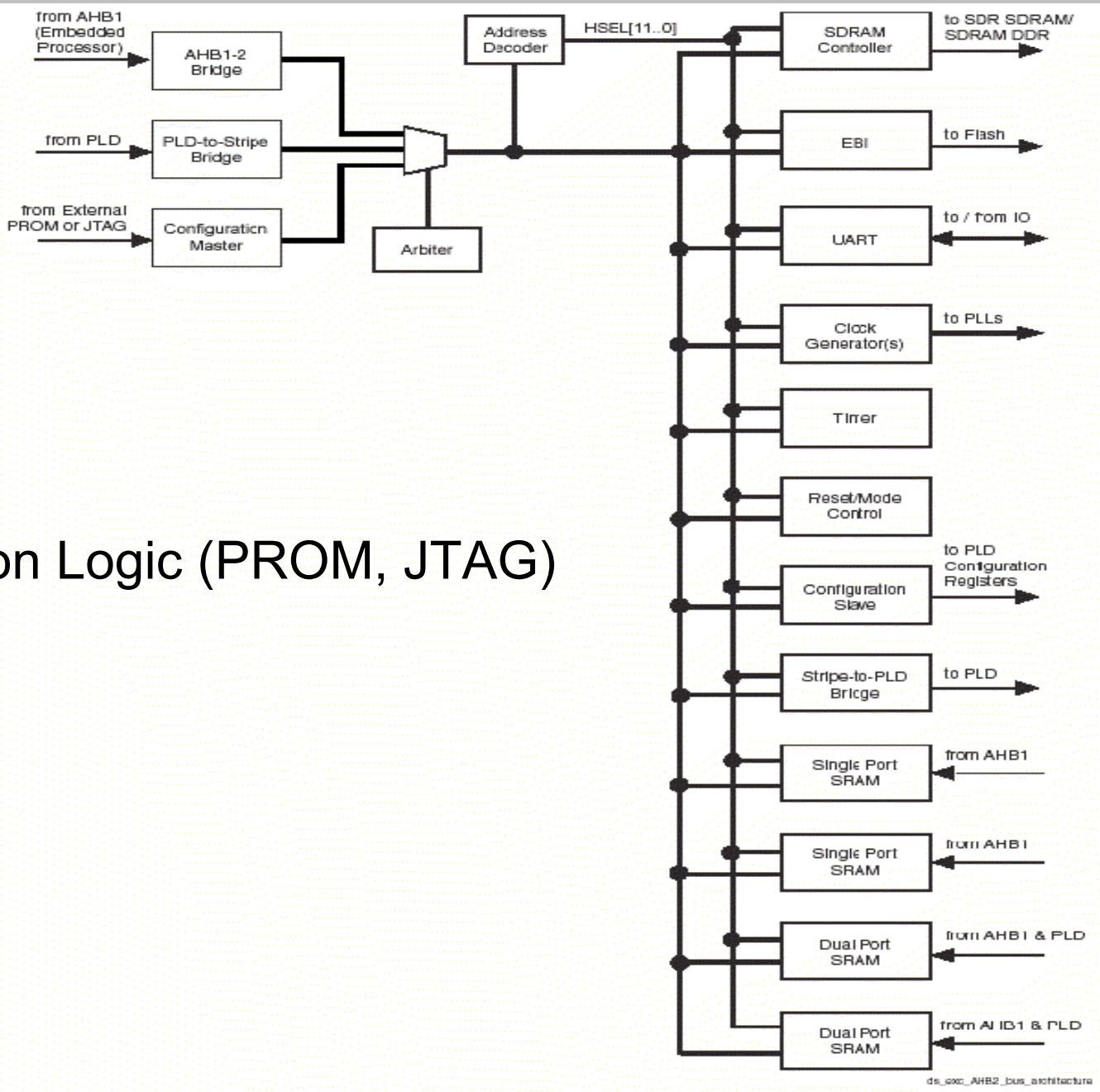
Internal / external Bus

- AHB1 :
  - ARM 922T
  - Interrupt Ctrl
  - Watch-Dog
  - SRAM/DPSRAM/
  - SDRAM Ctrl
  - Bridge → AHB2
- AHB2 :
  - Hardware stripe
  - PLL/Reset/Timer
  - UART, EBI
  - SDRAM Ctrl
  - AHB bridges ↔ PLD



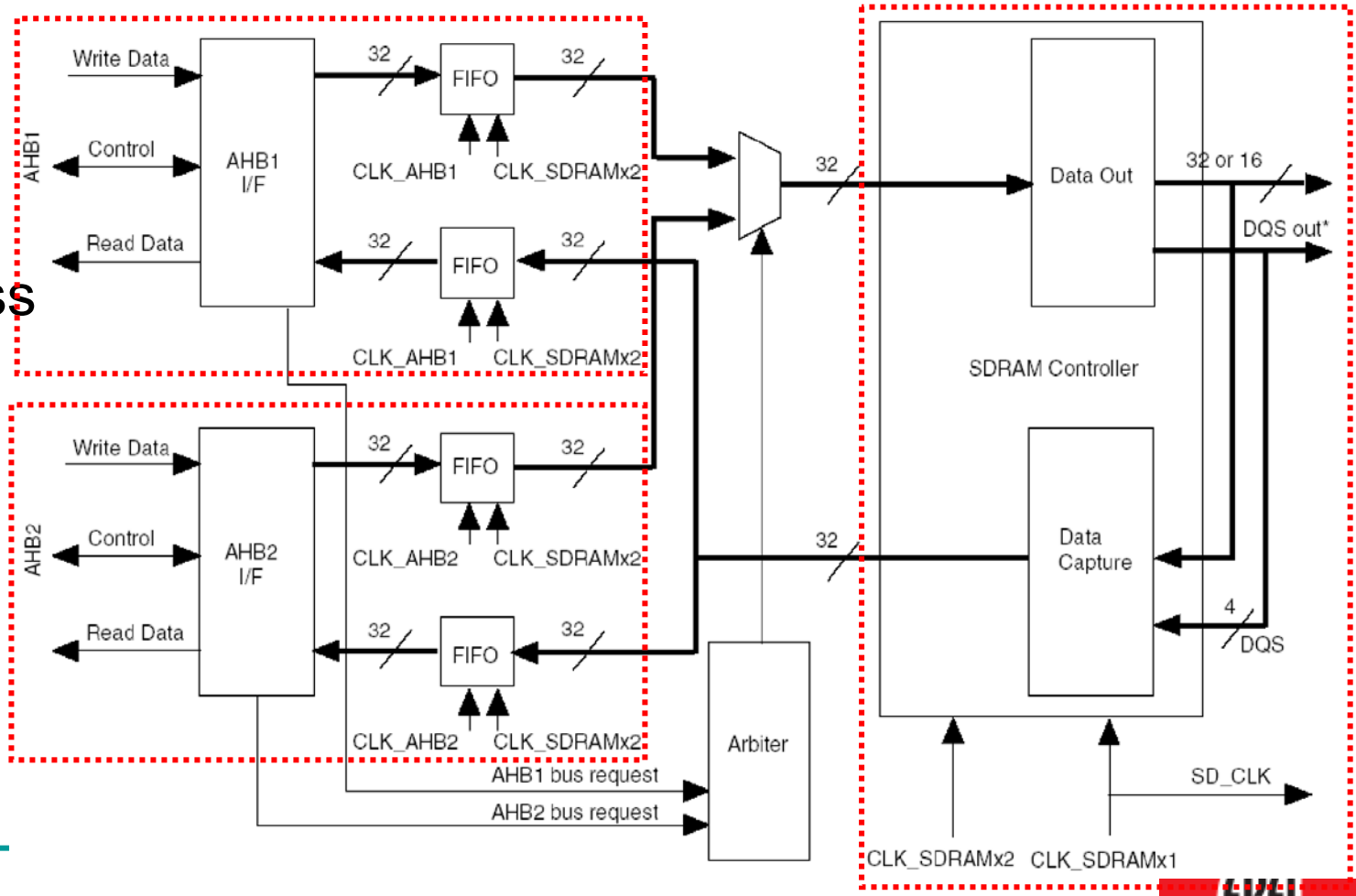
AHB2

- 3 Masters :
  - AHB1
  - Configuration Logic (PROM, JTAG)
  - PLD bridge
- 12 Slaves

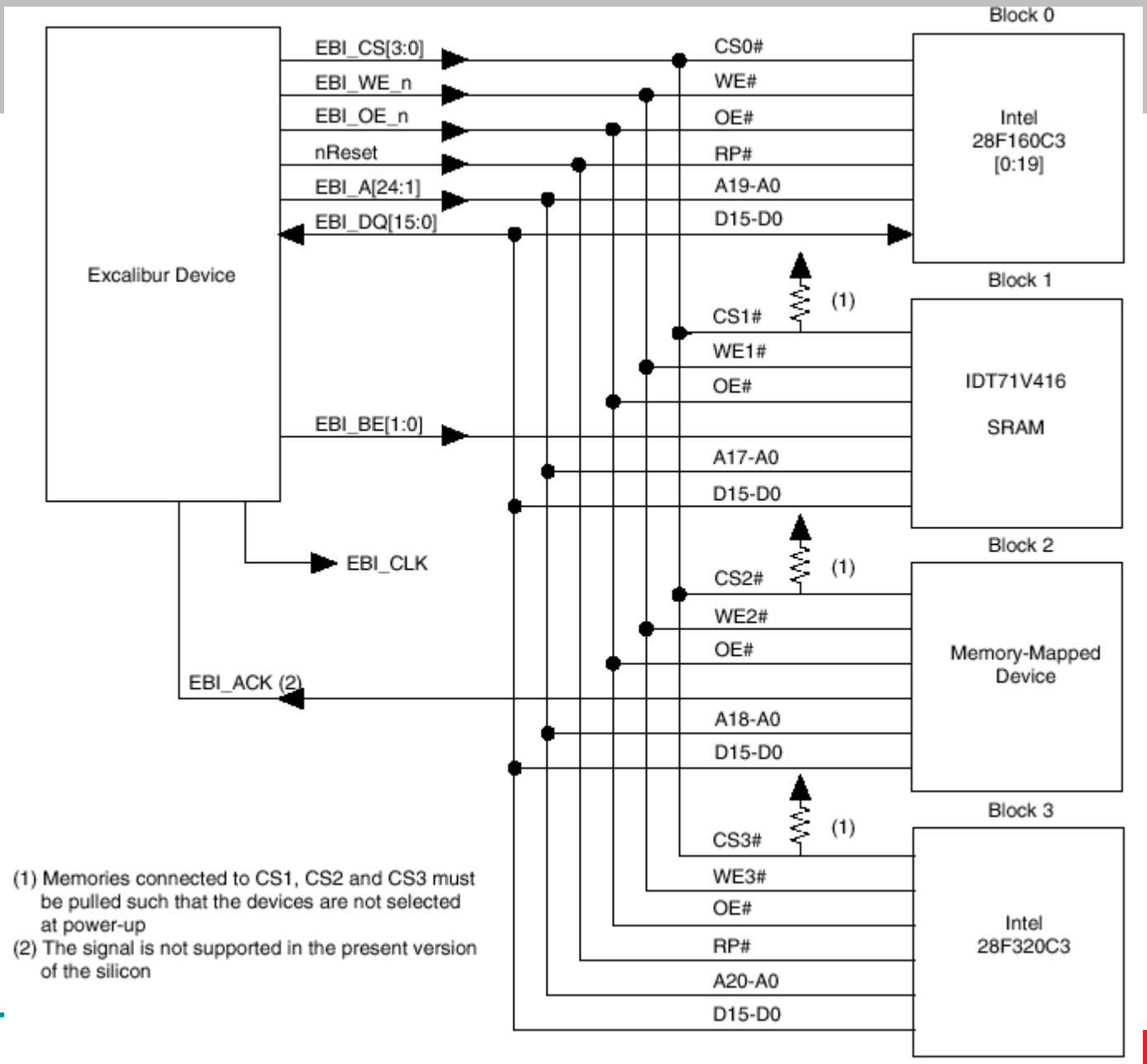


Ctrl DRAM

- 2 sources:
  - AHB1
  - AHB2
- FIFO access
- 1 arbiter

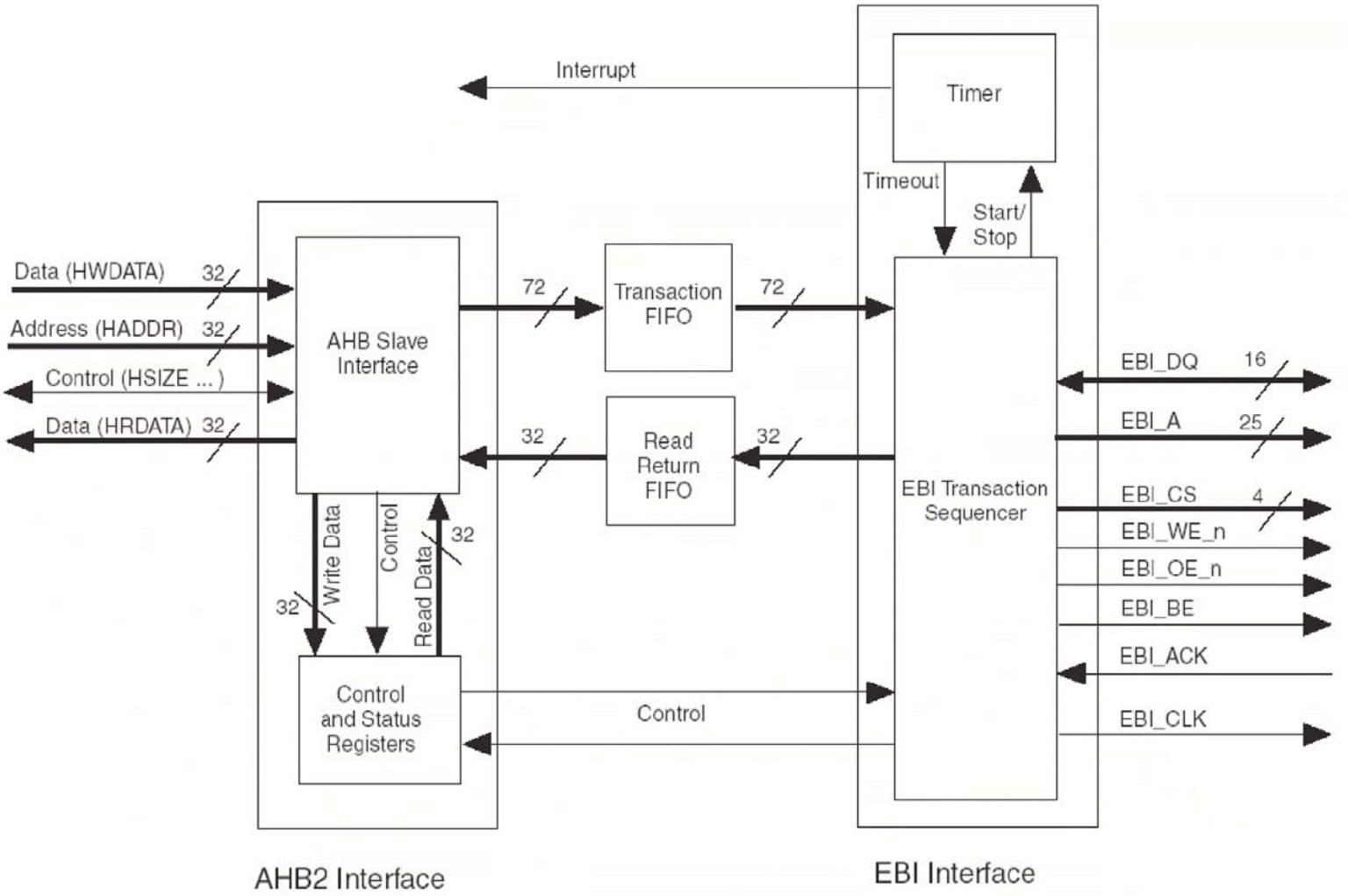


EBI



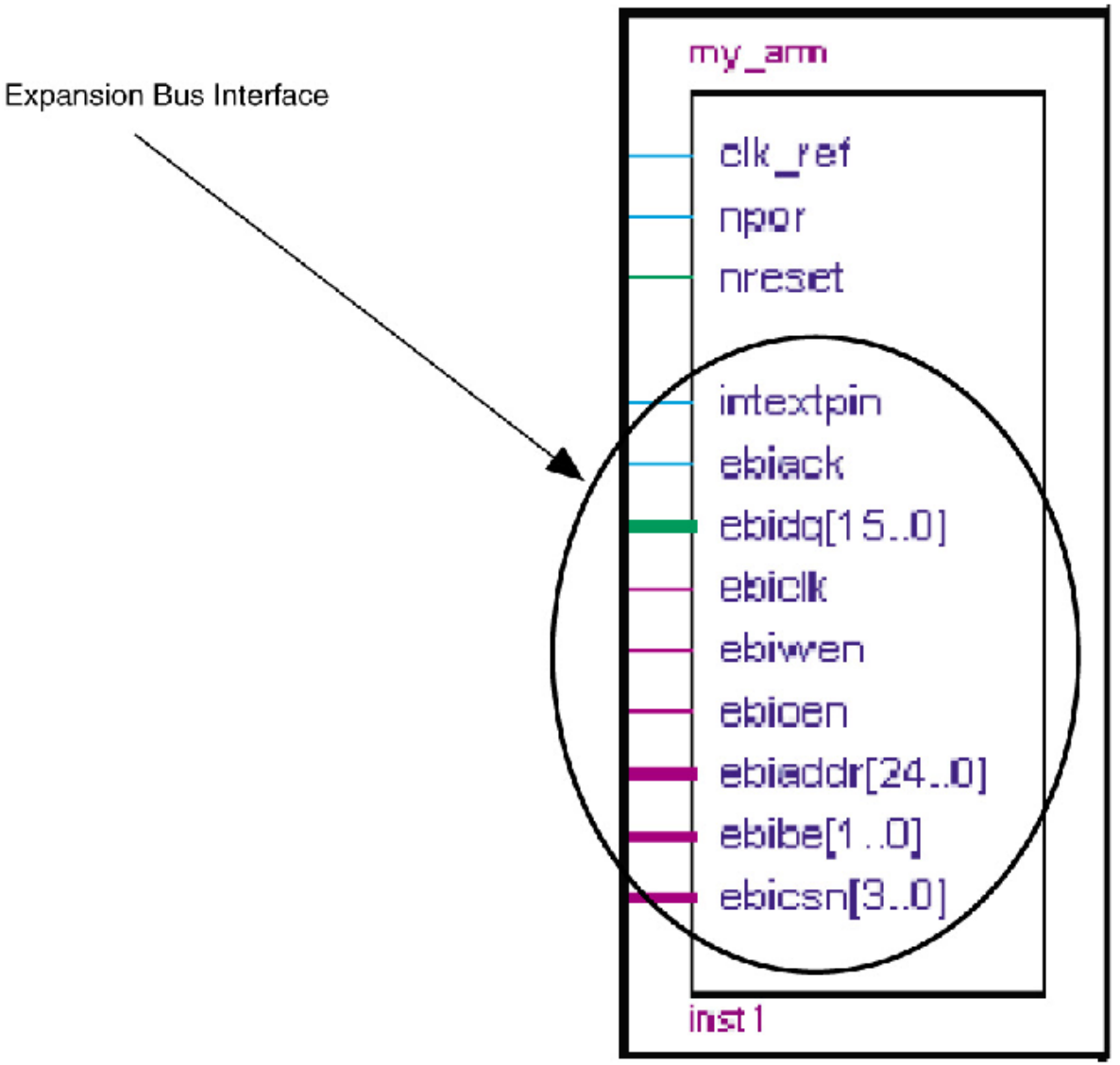
(1) Memories connected to CS1, CS2 and CS3 must be pulled such that the devices are not selected at power-up
(2) The signal is not supported in the present version of the silicon

Ctrl EBI



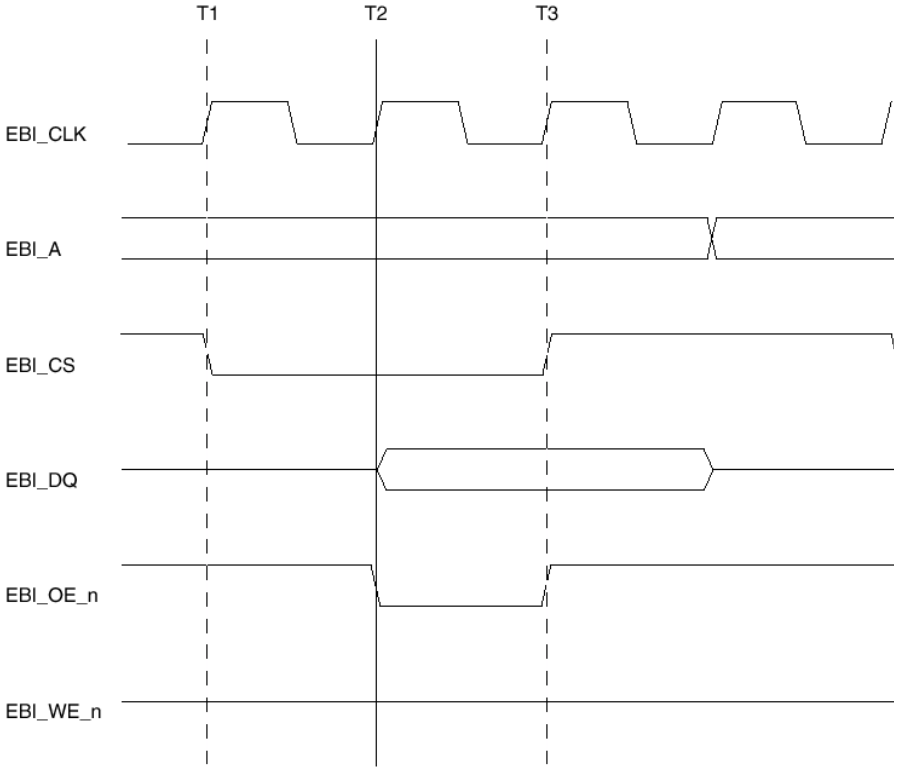
EBI

EBI Port on ARM Stripe Symbol

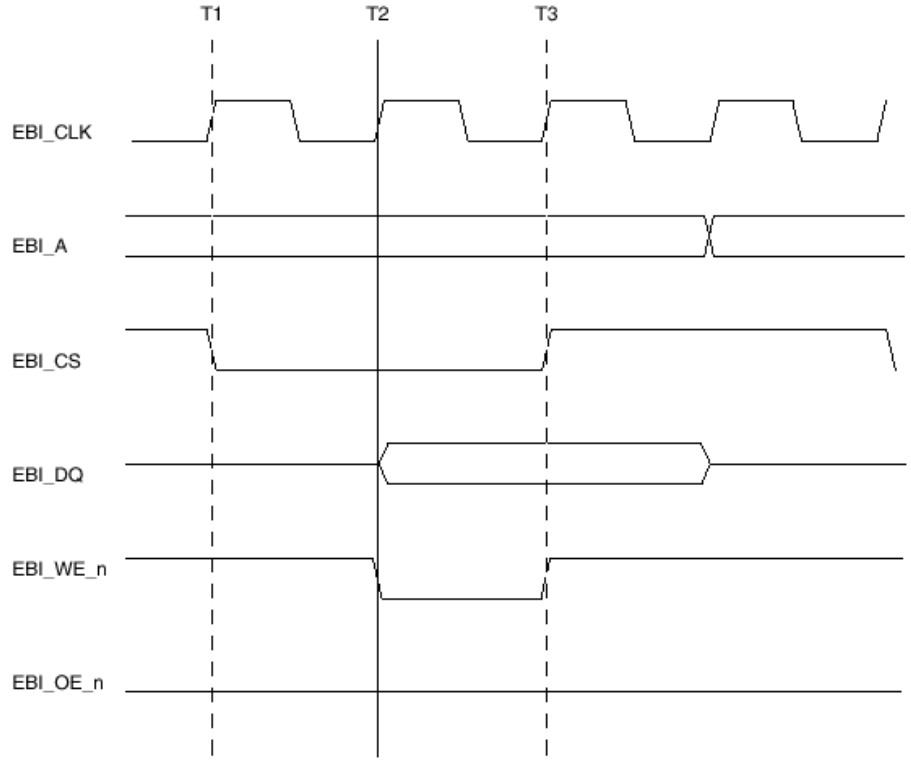


Cycles EBI, examples

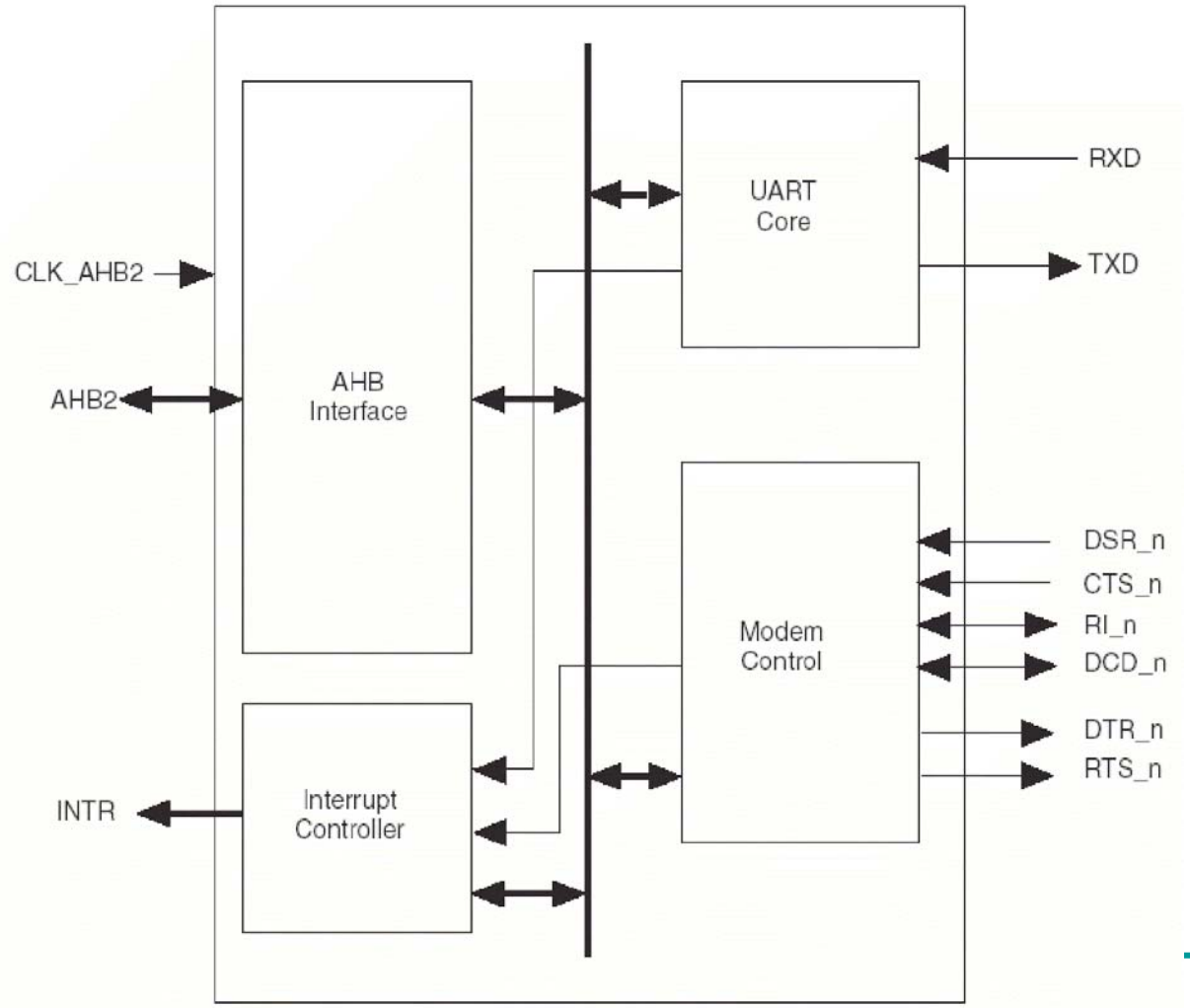
EBI Read 0 wait



EBI Write 0 wait

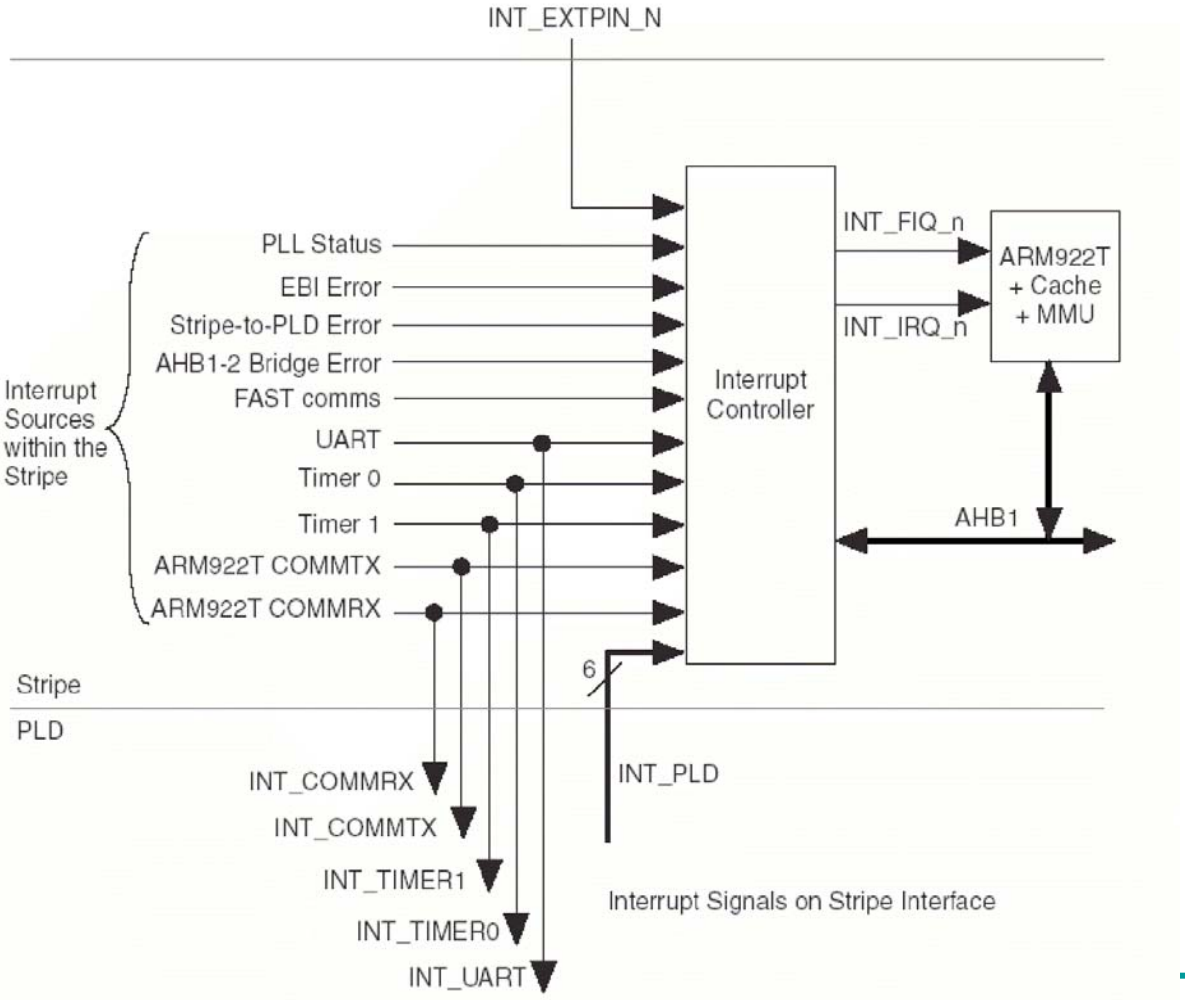


UART serial interface

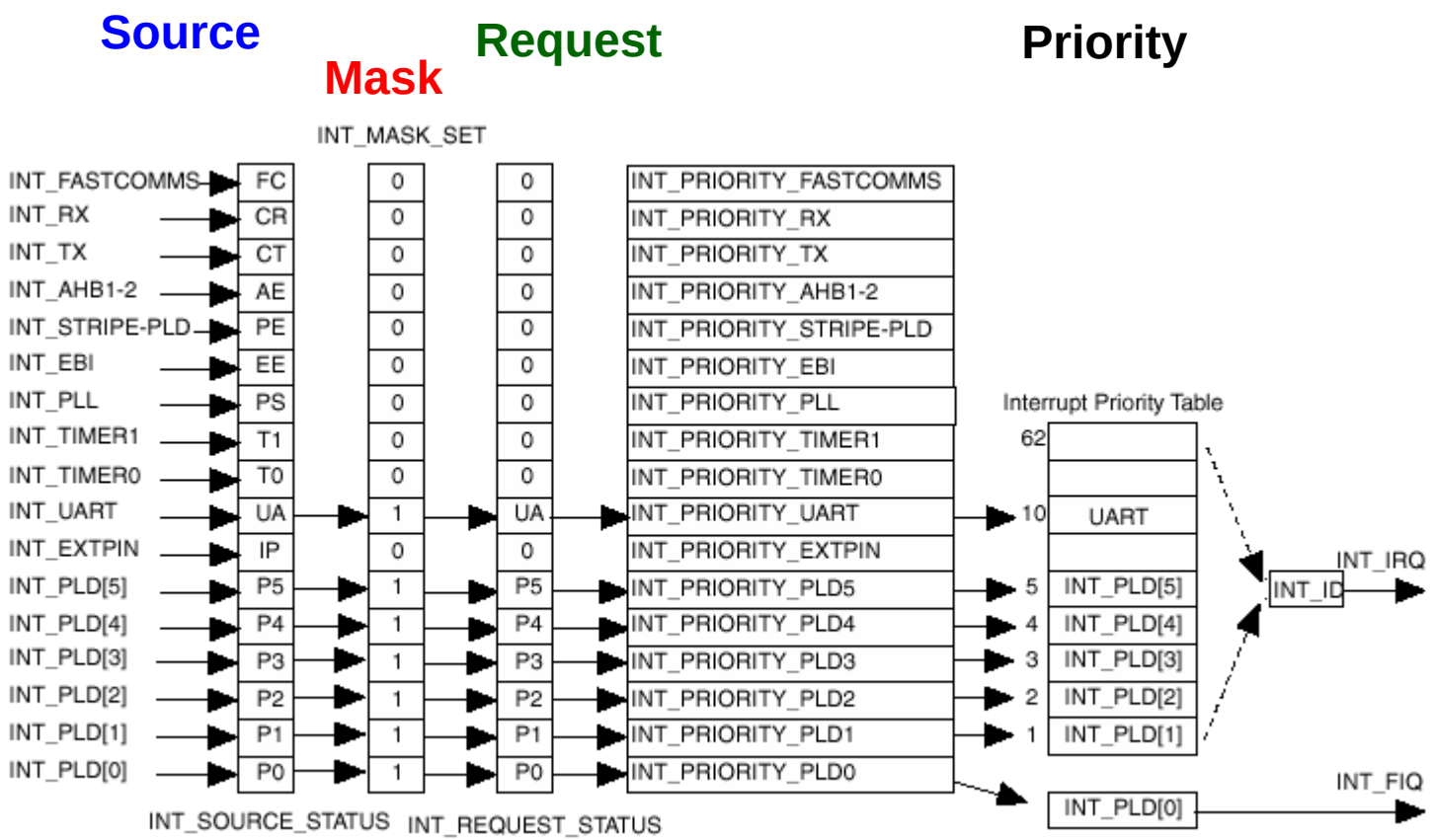


Interruptions

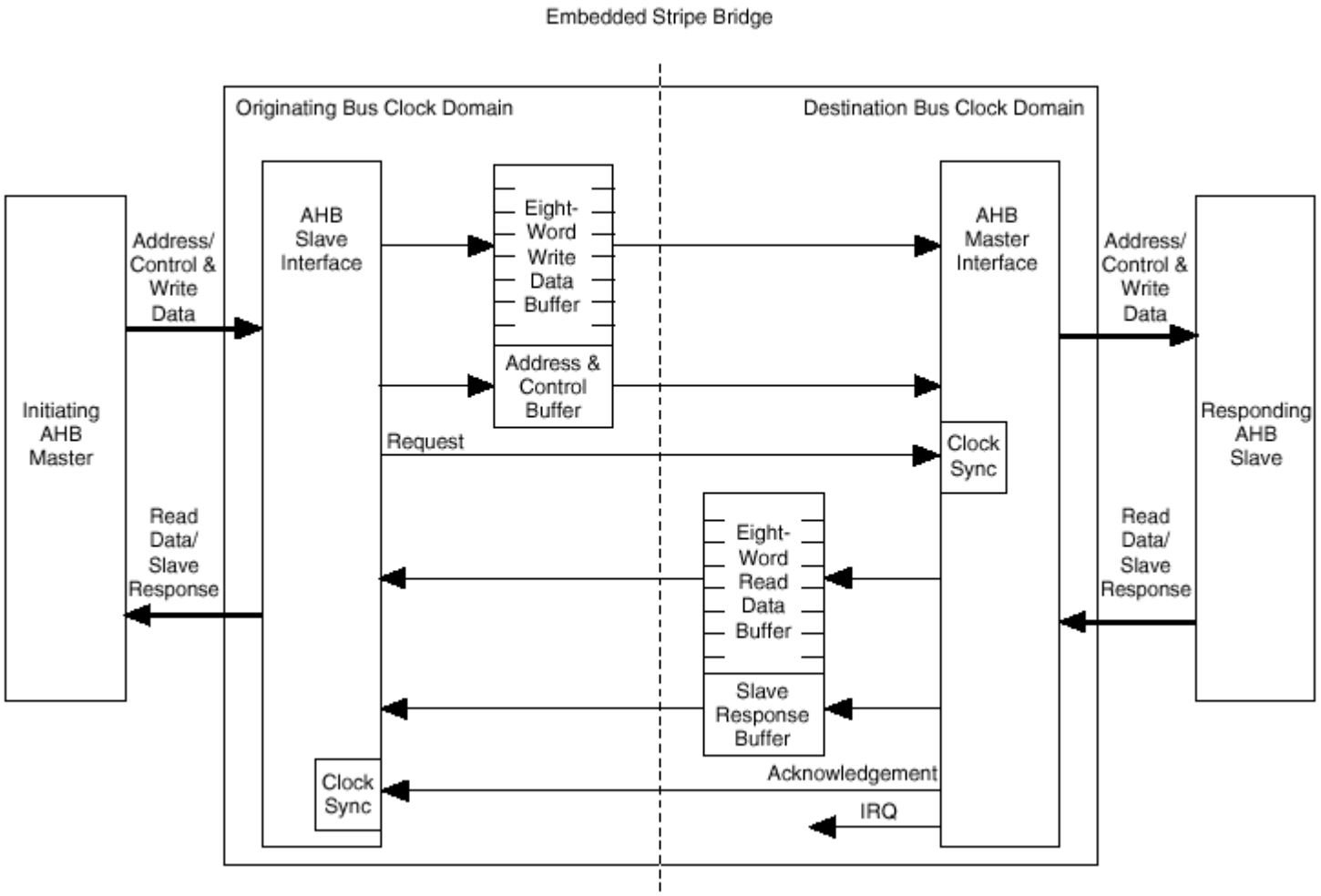
from AHB (11 sources)
or PLD (6 sources)



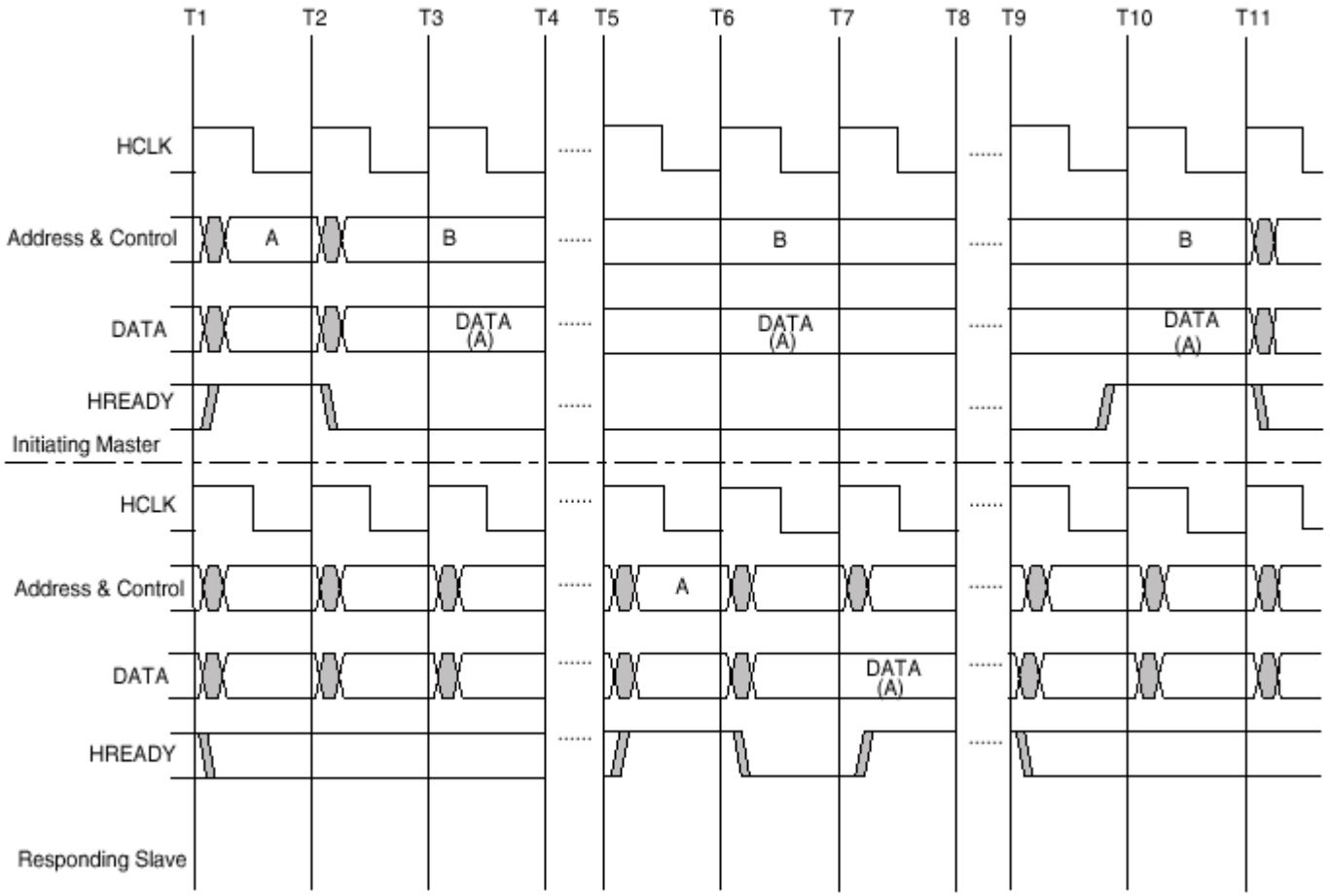
Interruptions Control registers



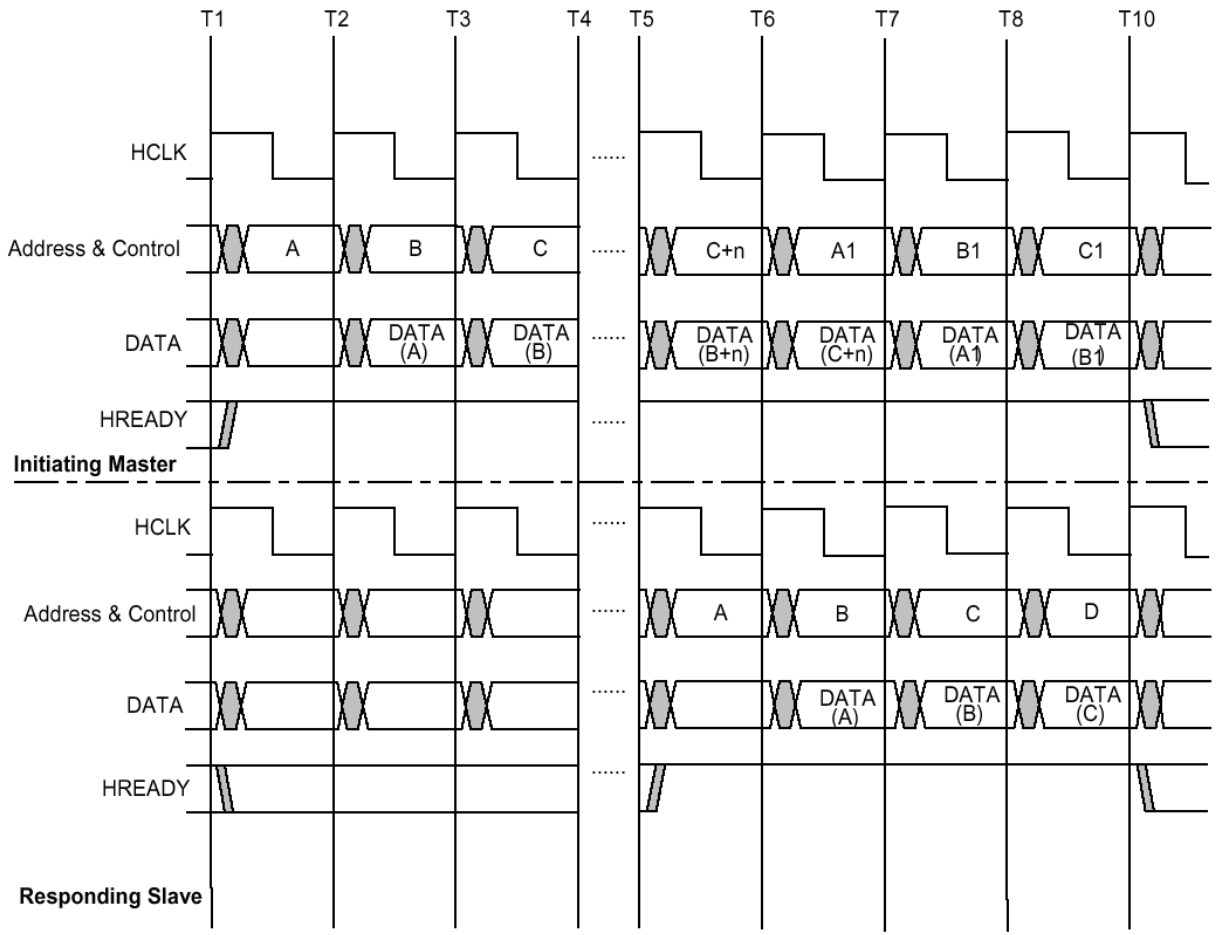
Stripe Bridge AHB1←→AHB2



Direct Writing



Writing through FIFO



Multi bridge, writing

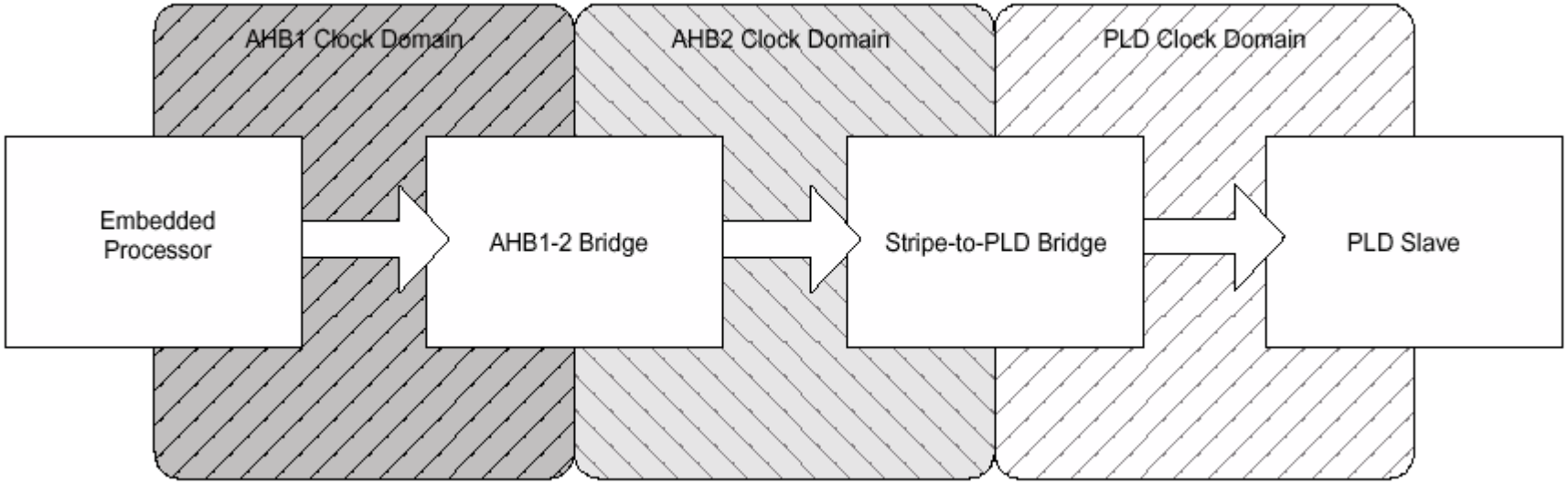


Table 1. Bridge Write Transaction Throughput from the Embedded Processor to a PLD Slave

Transaction Number	AHB1-2 Bridge	Stripe-to-PLD Bridge	Throughput (Mbytes per Second)
1	Write-posting enabled	Write-posting enabled	194.9
2	Write-posting enabled	Write-posting disabled	44.4
3	Write-posting disabled	Write-posting enabled	75.6
4	Write-posting disabled	Write-posting disabled	33

Multi-bridge, reading

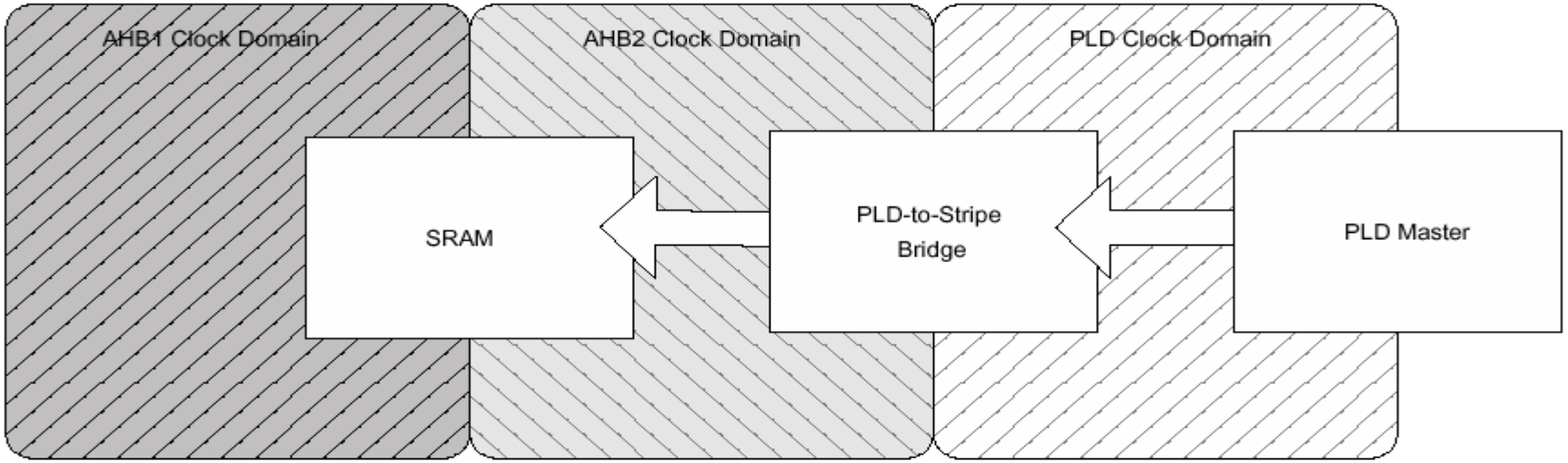
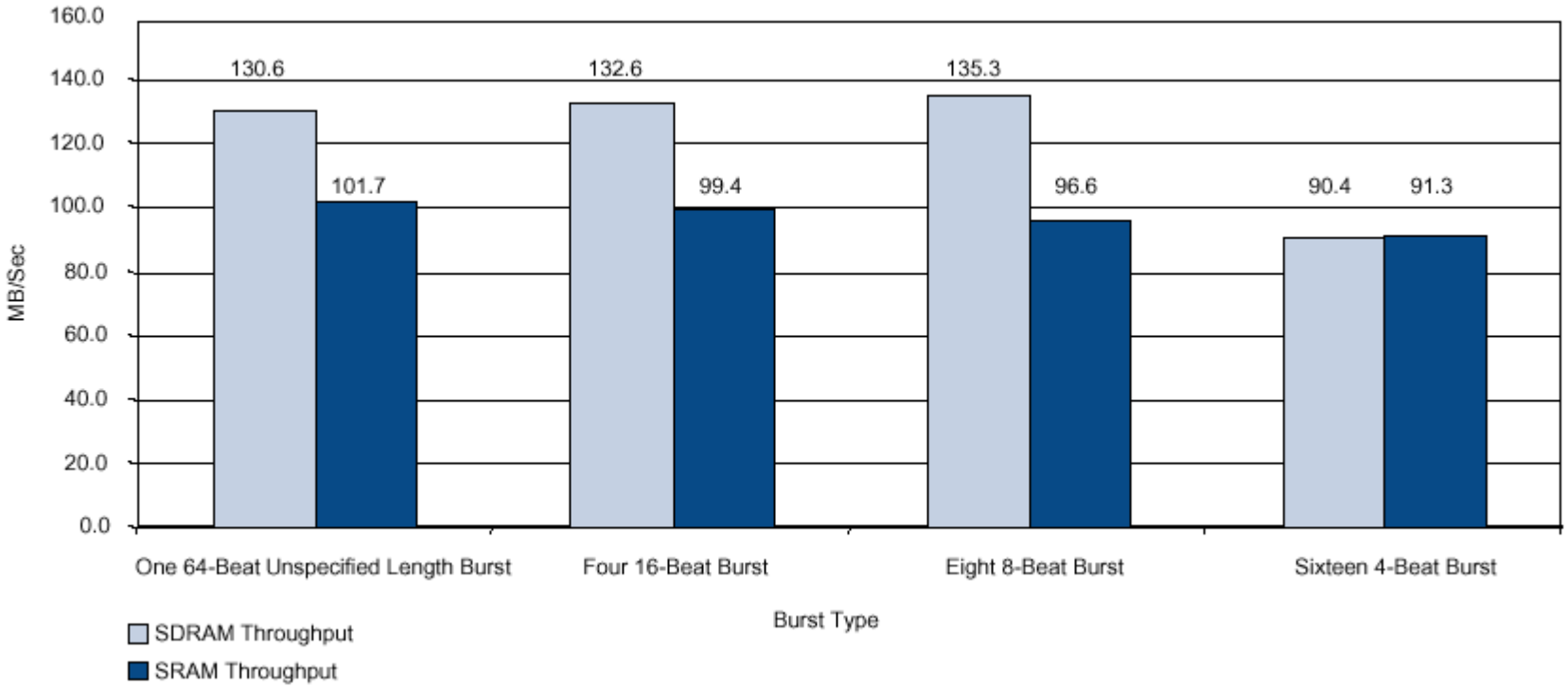


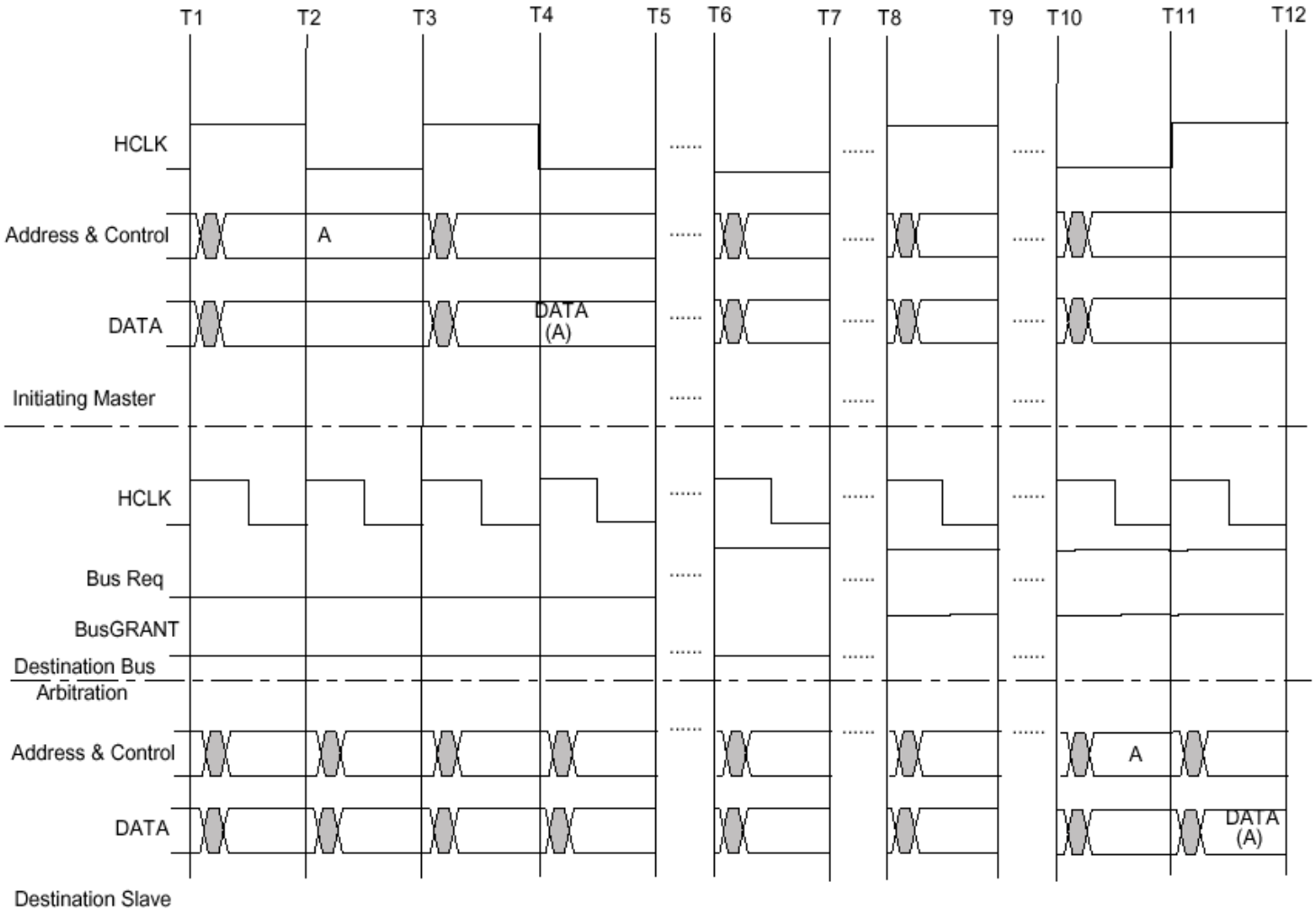
Table 2. Bridge Read Transaction Throughput from SRAM to a PLD Master

Transaction Number	Slave_hclk Frequency (PLD Clock Domain)	AHB2 Frequency	PLD-to-Stripe Bridge	Throughput (Mbytes per Second)
1	100 MHz	100 MHz	Pre-fetching enabled	100.4
2	100 MHz	100 MHz	Pre-fetching disabled	41.4
3	50 MHz	100 MHz	Pre-fetching enabled	98.5
4	50 MHz	100 MHz	Pre-fetching disabled	33.1

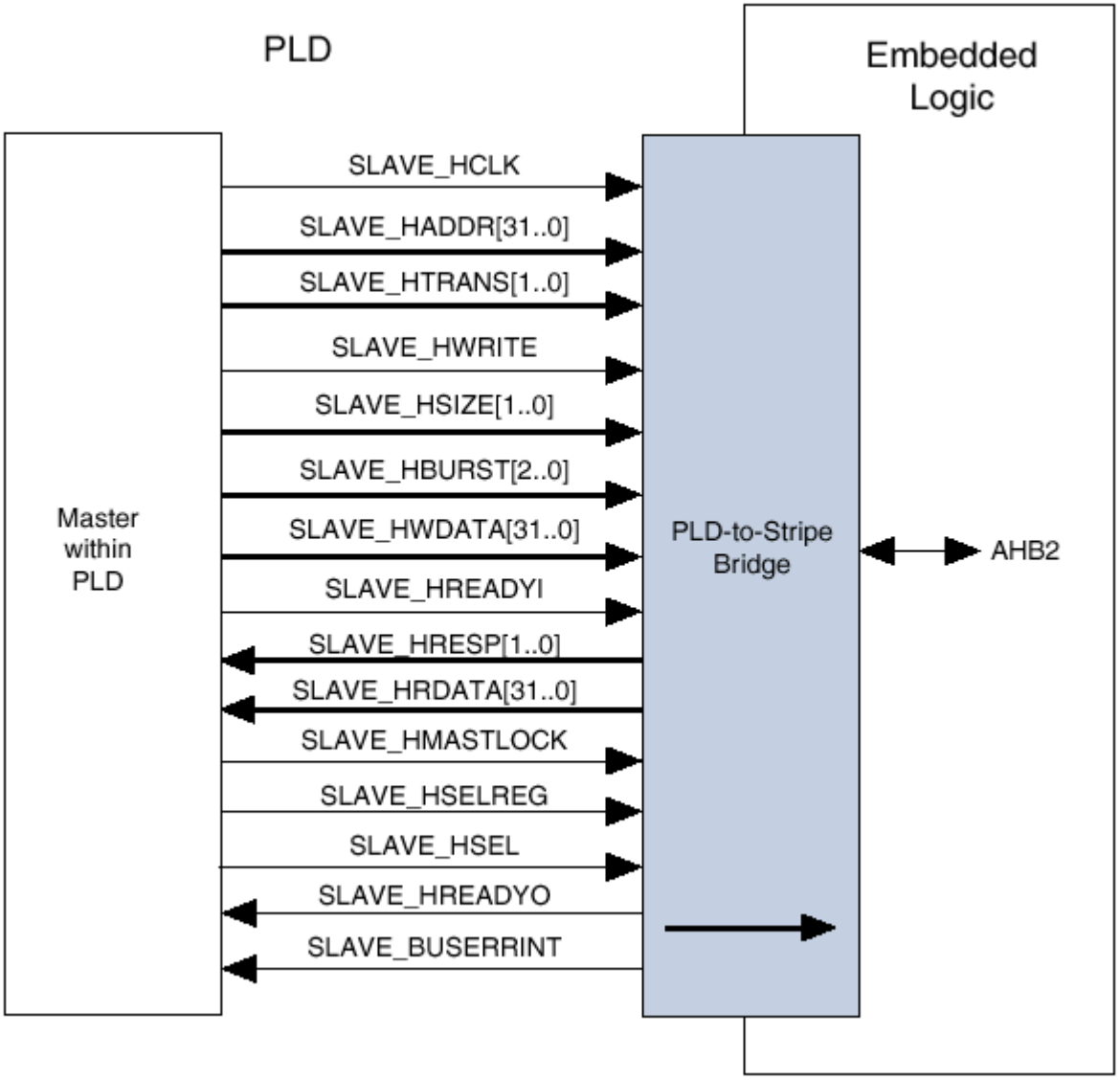
Burst transfers, comparison



Acces latency



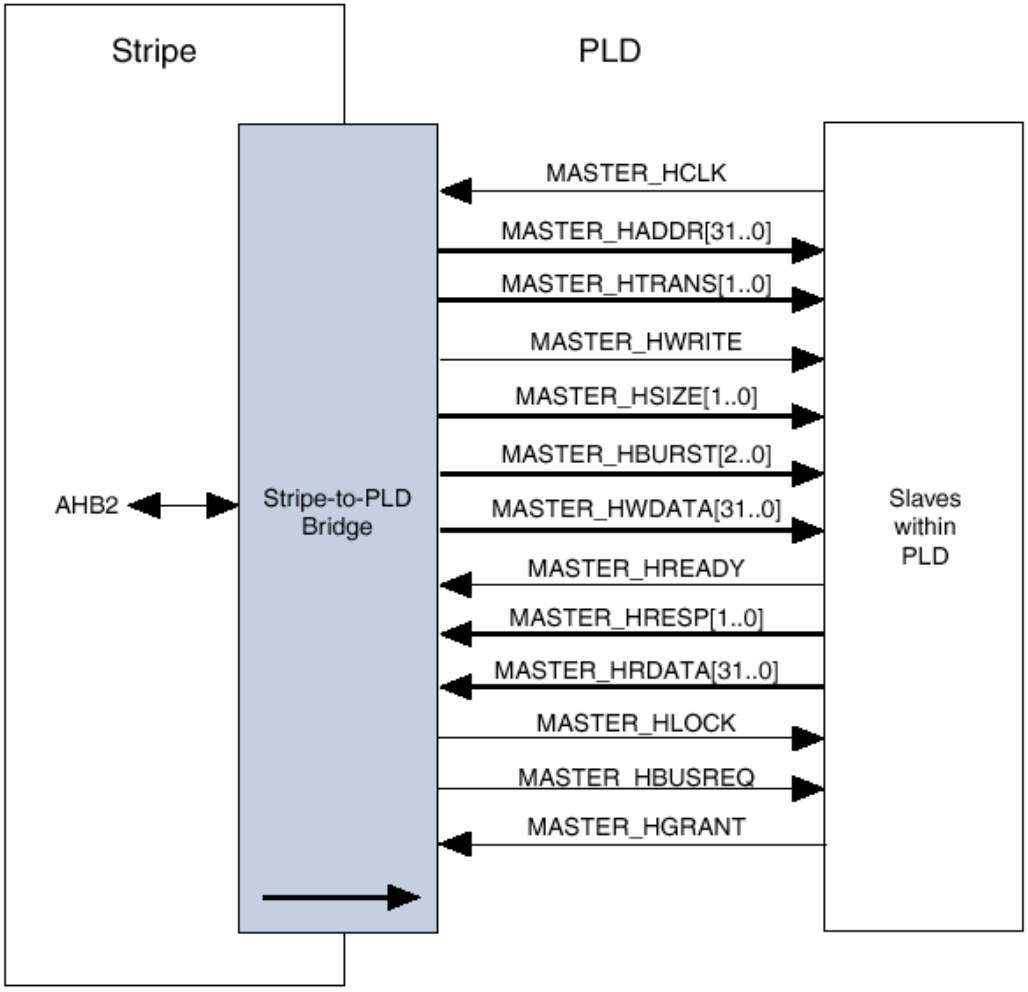
Master PLD → bridge AHB2



PLD → Stripe

Signal	Source	Description
SLAVE_HCLK	PLD	Times all bus transfers. Signal timings are related to its rising edge clock
SLAVE_HADDR[31..0]	PLD	32-bit system address bus
SLAVE_HTRANS[1..0]	PLD	The type of the current transfer
SLAVE_HWRITE	PLD	When high, this signal indicates a write transfer; when low, a read transfer
SLAVE_HSIZE[1..0]	PLD	Indicates the size of transfer
SLAVE_HBURST[2..0]	PLD	Indicates whether the transfer forms part of a burst
SLAVE_HWDATA[31..0]	PLD	Used to transfer data from the master to the bus slaves during writes
SLAVE_HREADYI	PLD	When high, this signal indicates that a transfer has finished on the bus. Slaves on the bus need SLAVE_HREADY as both an input and output signal
SLAVE_HREADYO	Stripe	When high, this signal indicates that a transfer has finished on the bus. Slaves on the bus need SLAVE_HREADY as both an input and output signal
SLAVE_HRESP[1..0]	Stripe	Additional information on the status of a transfer
SLAVE_HRDATA[31..0]	Stripe	Used to transfer data from bus slaves to the master during reads
SLAVE_HMASTLOCK	PLD	When high, indicates that the master requires locked access to the bus
SLAVE_BUSERRINT	Stripe	Interrupt signifying a bus error
SLAVE_HSELREG	PLD	Register selection signal
SLAVE_HSEL	PLD	Interface selection signal

Slave PLD ← Bridge



Stripe → PLD

Signal	Source	Description
MASTER_HCLK	PLD	Times all bus transfers.Signal timings are related to its rising edge clock
MASTER_HADDR[31..0]	Stripe	32-bit system address bus
MASTER_HTRANS[1..0]	Stripe	Type of the current transfer
MASTER_HWRITE	Stripe	When high, this signal indicates a write transfer; when low, a read transfer
MASTER_HSIZE[1..0]	Stripe	Indicates the size of transfer
MASTER_HBURST[2..0]	Stripe	Indicates whether the transfer forms part of a burst
MASTER_HWDATA[31..0]	Stripe	Used to transfer data from the master to the bus slaves during writes
MASTER_HREADY	PLD	When high, this signal indicates that a transfer has finished on the bus
MASTER_HRESP[1..0]	PLD	Additional information on the status of a transfer
MASTER_HRDATA[31..0]	PLD	Used to transfer data from bus slaves to the master during reads
MASTER_HLOCK	Stripe	When high, indicates that the master requires locked access to the bus
MASTER_HBUSREQ	Stripe	A signal from the master to the arbiter, requesting the bus

Stripe Registers

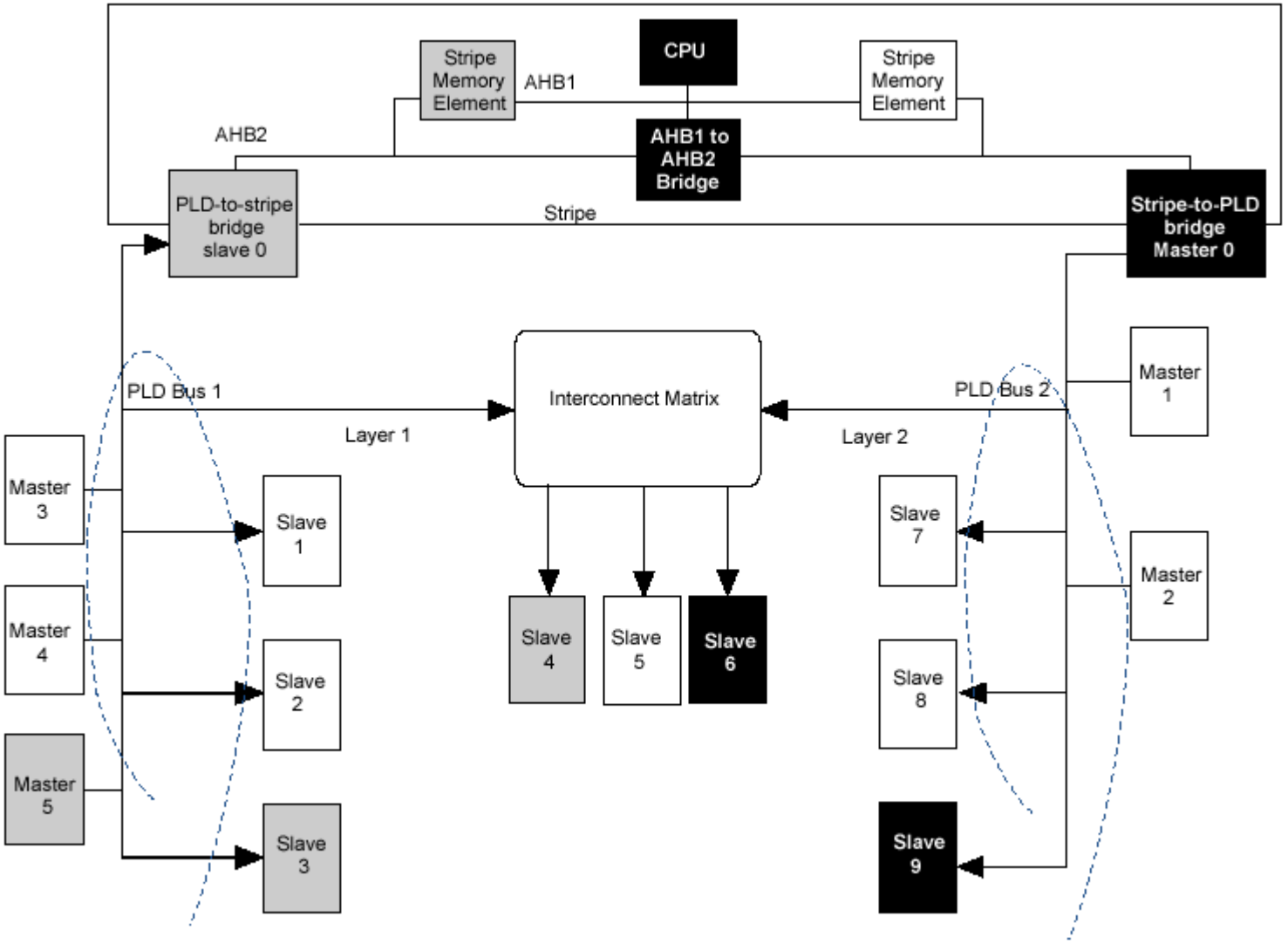
Configurations are specified by programmable interfaces

Register Name	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Address
AHB12B_CR																															NW	NP	Base + 100H
AHB12B_SR																								HBURST			HSIZE			HRTN	WF		Base + 800H
AHB12B_ADDRSR	HADDR																																Base + 804H
PLDSB_CR																															NW	NP	Base + 110H
PLDSB_SR																								HBURST			HSIZE			HRTN	WF		Base + 114H
PLDSB_ADDRSR	HADDR																																Base + 118H
PLDMB_CR																															NW	NP	Base + 120H
PLDMB_SR																								HBURST			HSIZE			HRTN	WF		innaccessible
PLDMB_ADDRSR	HADDR																																innaccessible

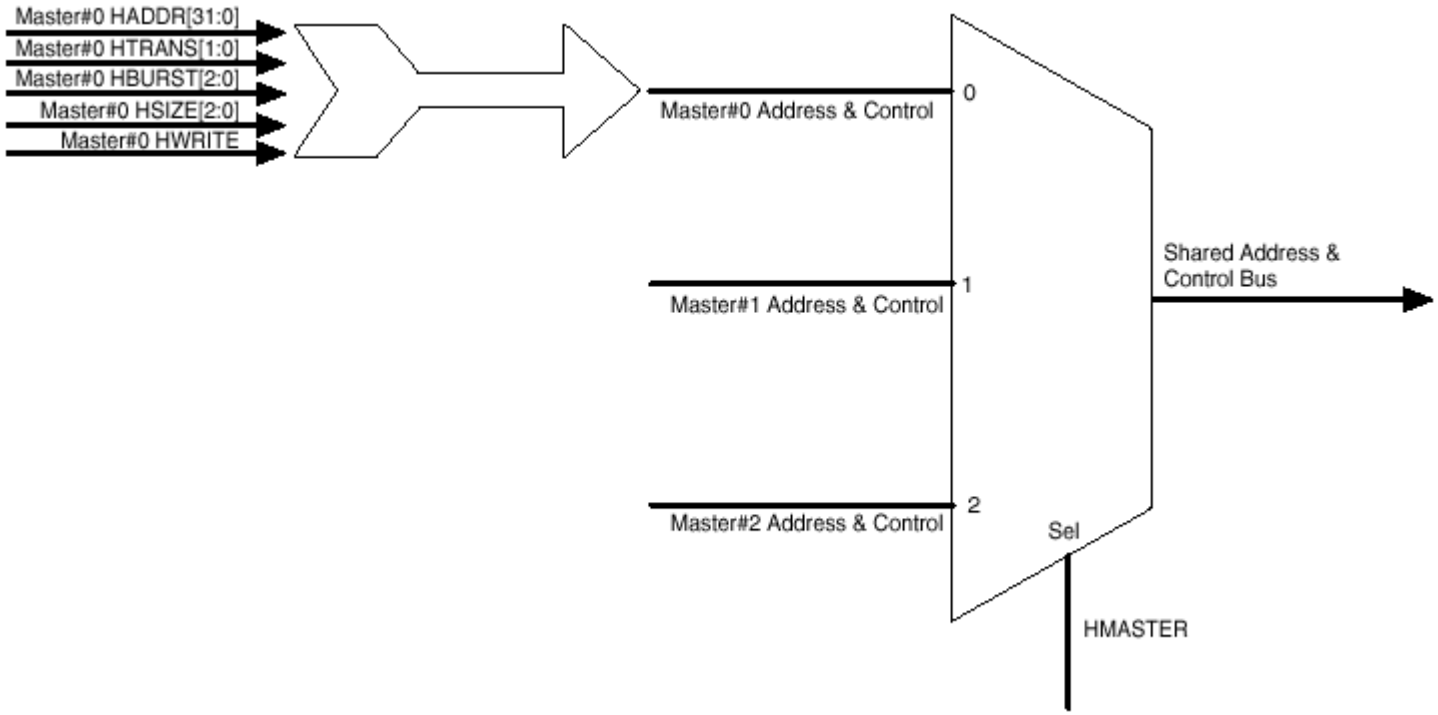
Registre Memory Map

Register Name	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Address
MMAP_xxx																															NW	NP	Base + nnn

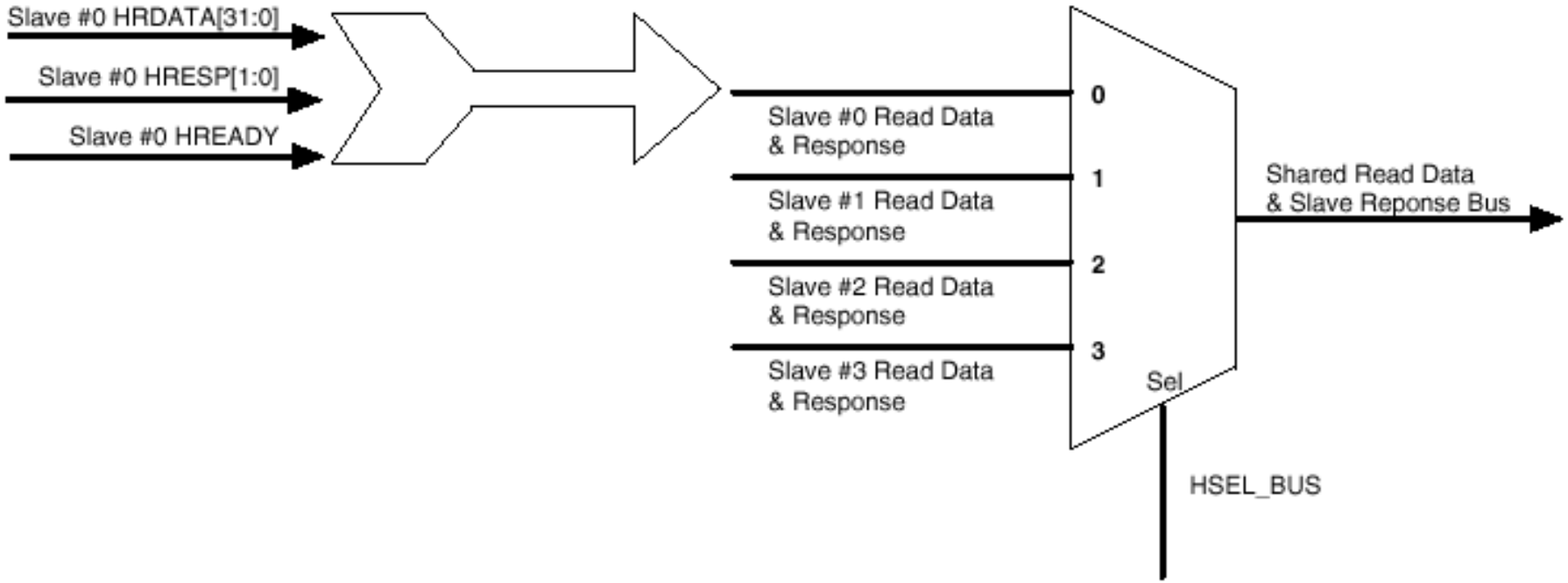
Multi-master, example



Multiplexed Bus
Address + ctrl

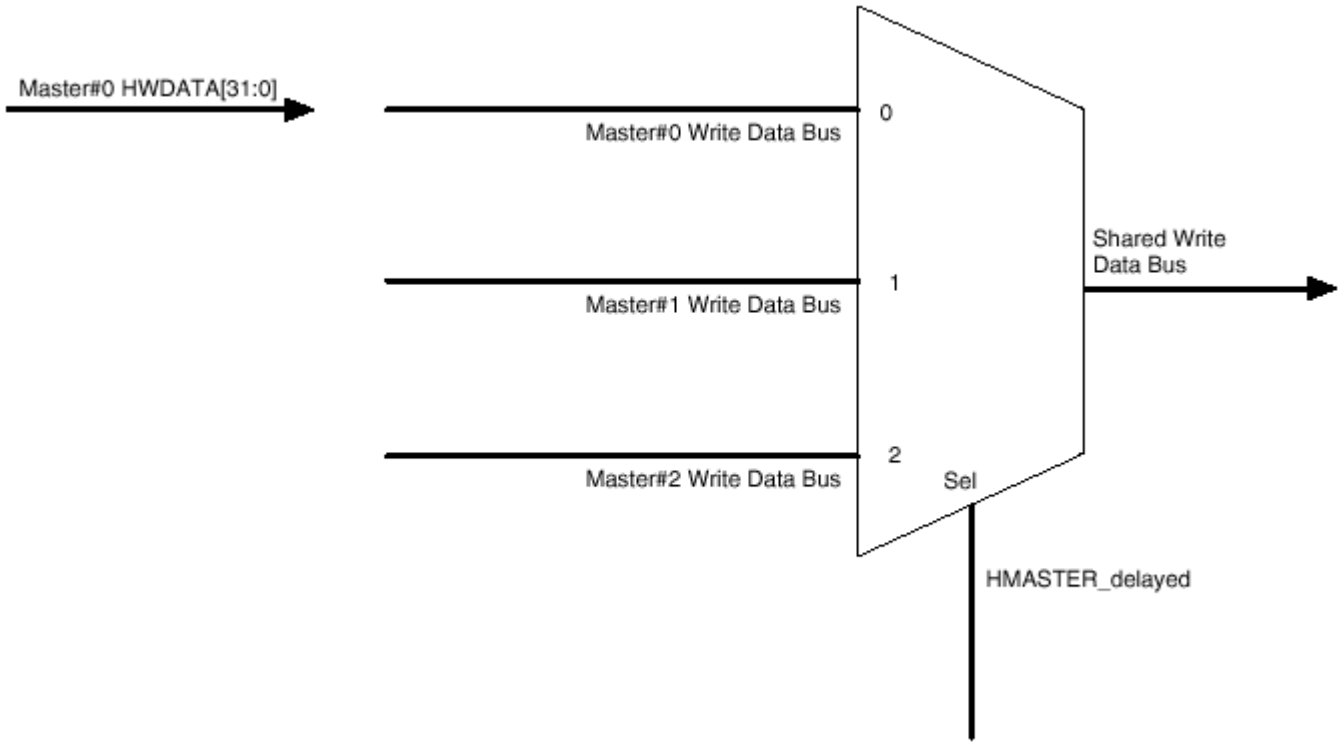


Multiplexed Bus,
data read → Masters



Multiplexed Bus,

Masters → data write



PLD interface

- Hardware bridge : AMBA
- NIOS processors → Avalon bus
- Need AMBA ←→ Avalon bridge

- Use of SOPC Builder™ :
  - ➢Automatic system generation
  - ➢Cross Development for software