

ARM810: Dancing to the Beat of a Different Drum

Guy Larri



**90 Fulbourn Road, Cherry Hinton,
Cambridge, CB1 4JN, England
Guy.Larri@arm.com**

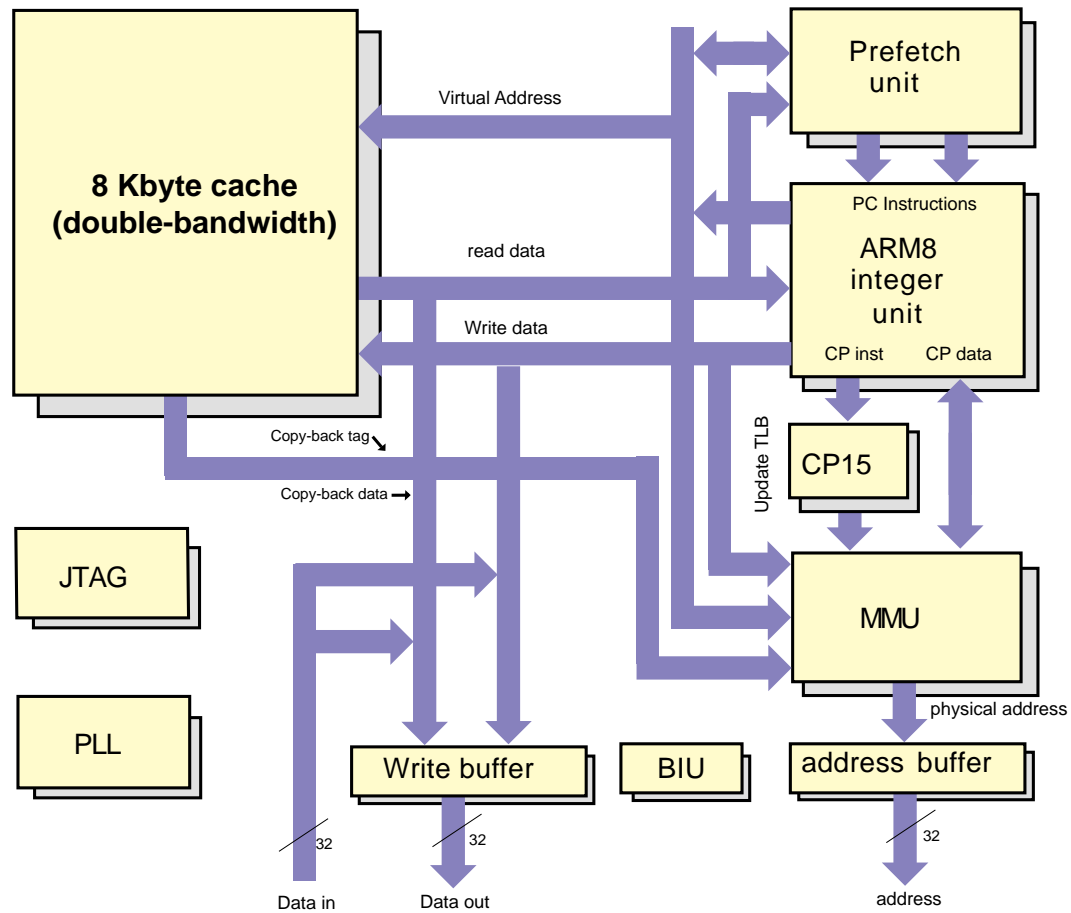
Commercial Drivers For ARM810

- **As always, need more Performance**
 - Twice the performances of ARM710 on same process
- **Embedded, low power, portable applications, requires:**
 - Inexpensive = small die, small plastic package, limited pin count
 - Very Focused on Low Power
- **Licensing Business Model Requires:**
 - Performance increase on commodity processes
 - Low power
 - Portable to commodity 0.6 μ m/0.5 μ m 3.3V 3-Layer Metal CMOS processes, migration path to 0.35 μ m
 - Modular Design: ARM8 Integer Core is separate product
 Cache size variants
 Variant with no MMU
 - Use as an embedded macrocell and stand-alone cached processor.

ARM810 ...

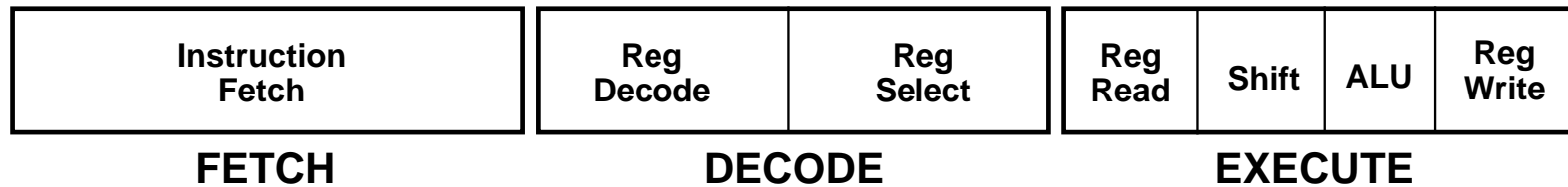
Architecture	ARMv4 Processor
CPU	5 stage pipeline + static branch prediction
Cache	8kB Unified Cache Write-Through and Copy-Back
TLB	64 entry TLB, 3 Mapping sizes
Performance	84 Dhrystone MIPs @ 72MHz
Power	0.5W @ 3.3V
Die Size	53.5mm ² (not including pad ring)
First Process	CMOS 3-layer metal, 0.6μm drawn 0.5μm drawn gates
Portable to	Commodity 0.6μm/0.5μm/0.35μm CMOS
Package	144 TQFP
Nice features	Integrated PLL, Lockable Cache/TLB
Markets	PDA, Network Computer

ARM810 Block Diagram

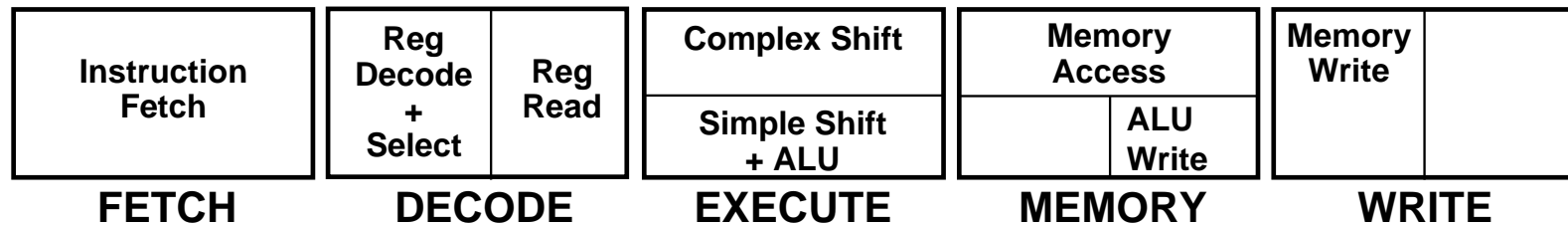


Pipeline changes for ARM8

ARM 7 Pipeline



ARM 8 Pipeline



Integer Core CPI Improvement

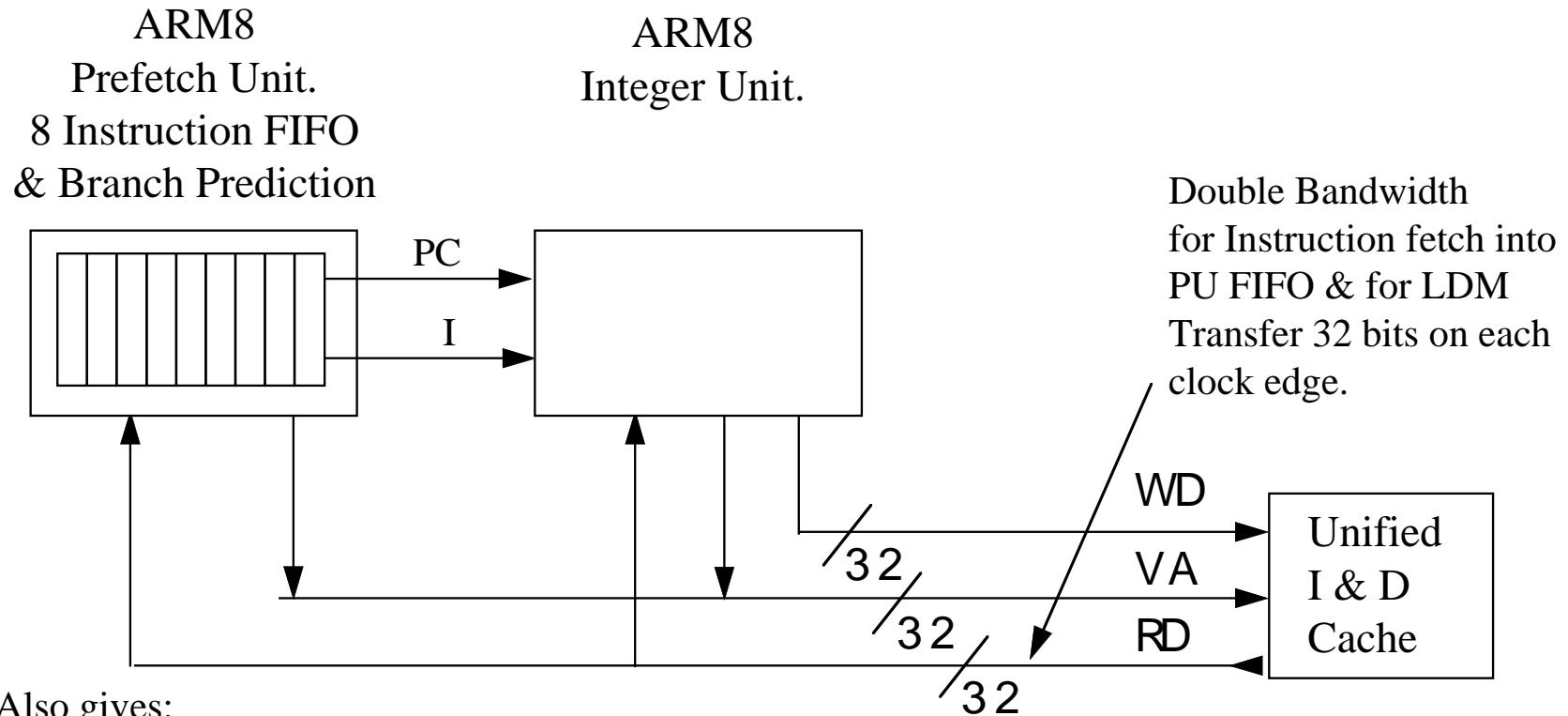
ARM8 CPI ~ 1.4

ARM7 CPI ~ 1.9

Improvement achieved by

Feature	% Improvement over ARM7
Single Cycle LDR	~ 12 %
Single Cycle STR	~ 4 %
Double bandwidth LDM	~ 6 %
Static Branch Prediction	~ 10 %
Total	~ 35 %

Satisfying Bandwidth Requirement



Also gives:

Single memory port for ROM/SRAM (Cache-less) systems.

Small number of Buses

Cycle Count Summary

Branch	min 0	Correctly Predicted		
	max 3	Incorrectly or Not Predicted		
Branch and Link	min 1	Correctly Predicted		
	max 3	Not Predicted		
Multiply and Accumulate	min 3	32 x 8 → 32		
	max 7	32 x 32 + 64 → 64		
	Normal case	Complex Operand Shift *	3rd read operand	Write to PC
Logical	1		+1	+2
Add, Subtract	1	+1	+1	+2
Load Word, Half, Byte	1	+1		+4
Store Word, Half, Byte	1		+1	
Load Multiple Words	$\lceil n/2 \rceil + 1$	where n = #registers		+4
Store Multiple Words	n	where n = #registers		

* = Shift other than LSR by 0, 1, 2 or 3 bits

Cache Features

- **8kB Unified Instruction and Data Cache**
- **4 Words per Cache Line, 64-way associative**
- **Random Replacement Algorithm**
- **Cache supports Copy-Back and Write-Through operation**
 - Selectable per-page in page-table entry
 - Copy-Back reduces write traffic to main memory
 - more main memory bandwidth available for DMA
 - lower system power
 - Write-Through good for frame buffers and easy upgrade from ARM710
- **Flexible Cache & TLB locking for real time applications**
 - Cache contents can be locked with granularity of 128 bytes
 - Gives low interrupt latency / guaranteed execution time for real-time applications

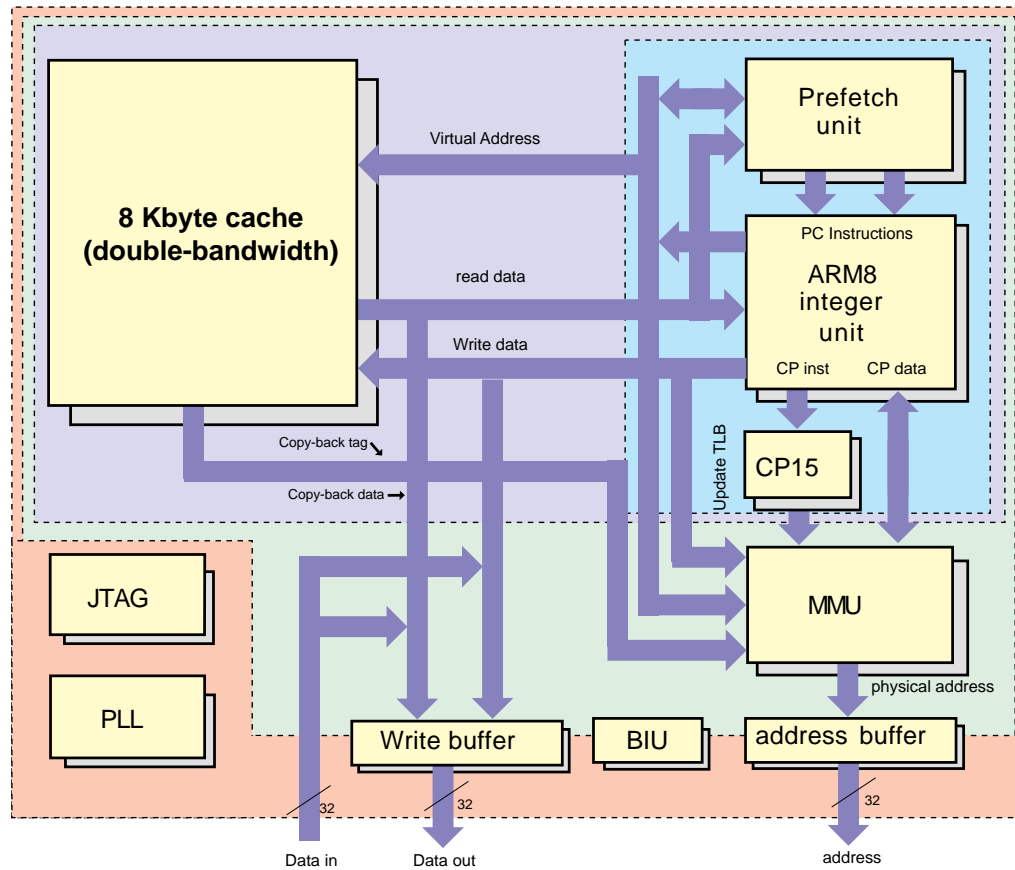
Cache Implementation

- **Cache is virtually addressed → low power**
 - No address translation required for cache read-hits
 - No address translation required for cache write-hits to copy-back regions
- **Cache stores only virtual tags**
 - Translate addresses for cache-line cast-outs when they occur
 - Avoids storing 512 lines * 25 bits = 1.6kB of physical tags
- **Cache implemented from 1kB CAM-RAM segments**
 - Only 1 segment active for each access
 - Segment selected by 3 bits of Virtual Address
 - Easy to build cache size variants
- **Double bandwidth read port to ARM8**

ARM810 μ Architecture Design Style for Low Power

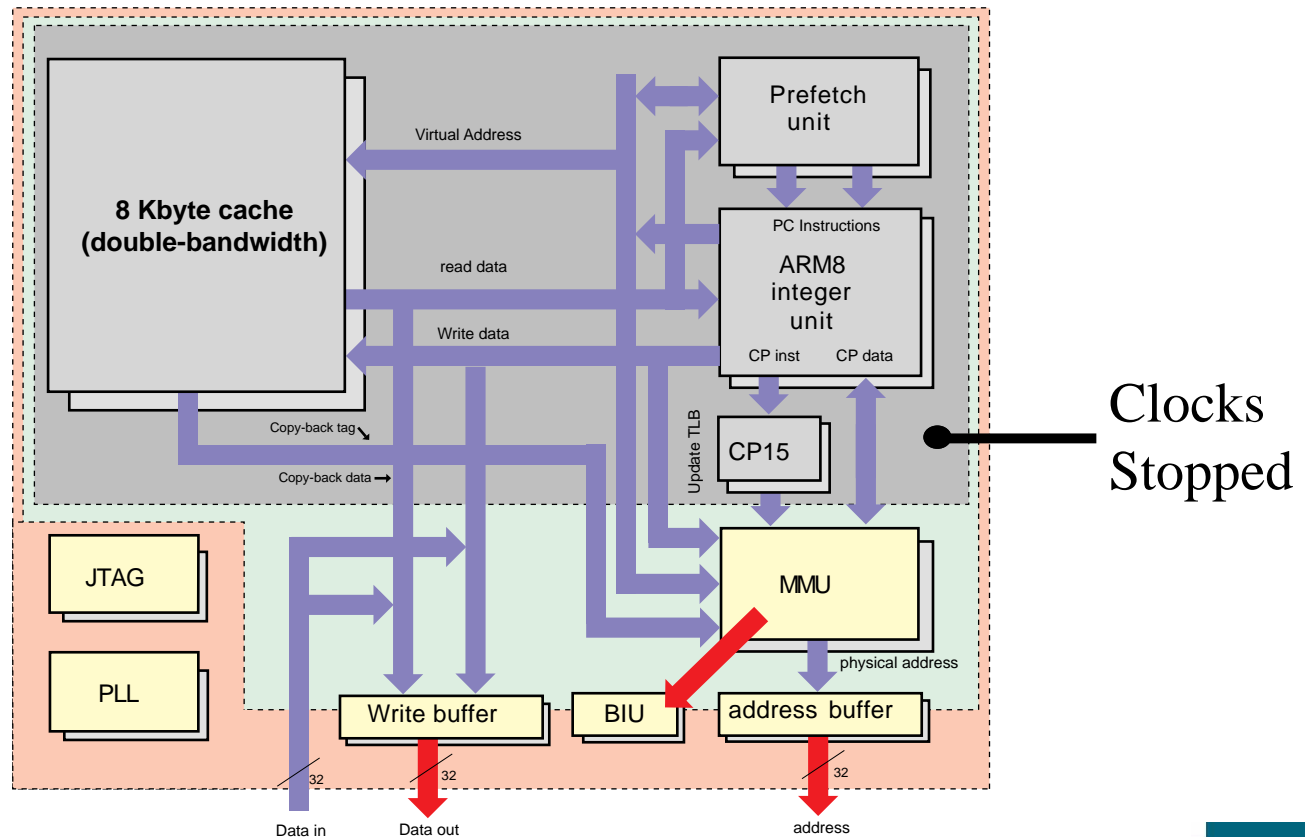
- **Hierarchy of Clocking Domains**
 - Stop clocks to as much of chip as possible for each stall type or off-chip access.
e.g. TLB miss stops clocks to Cache and ARM8 until page-table walk completes.
When TLB requests use of bus, bus controller will stop clock to TLB until write buffer empties, making the bus available to the TLB.
- **Separate controllers for each clocking domain.**
 - Yields modular control logic
e.g. No change to cache control or to ARM8 if MMU is removed.
- **Pipelined Cache and Bus Controller**
 - Maintain 1 write per cycle into cache and write-buffer while giving multiple cycles to resolve all controller scenarios.
 - Yields low power via clean synchronous signal transitions on pins.
 - Yields optimal use of sequential bursting on external bus.

ARM810 Clocking Domains



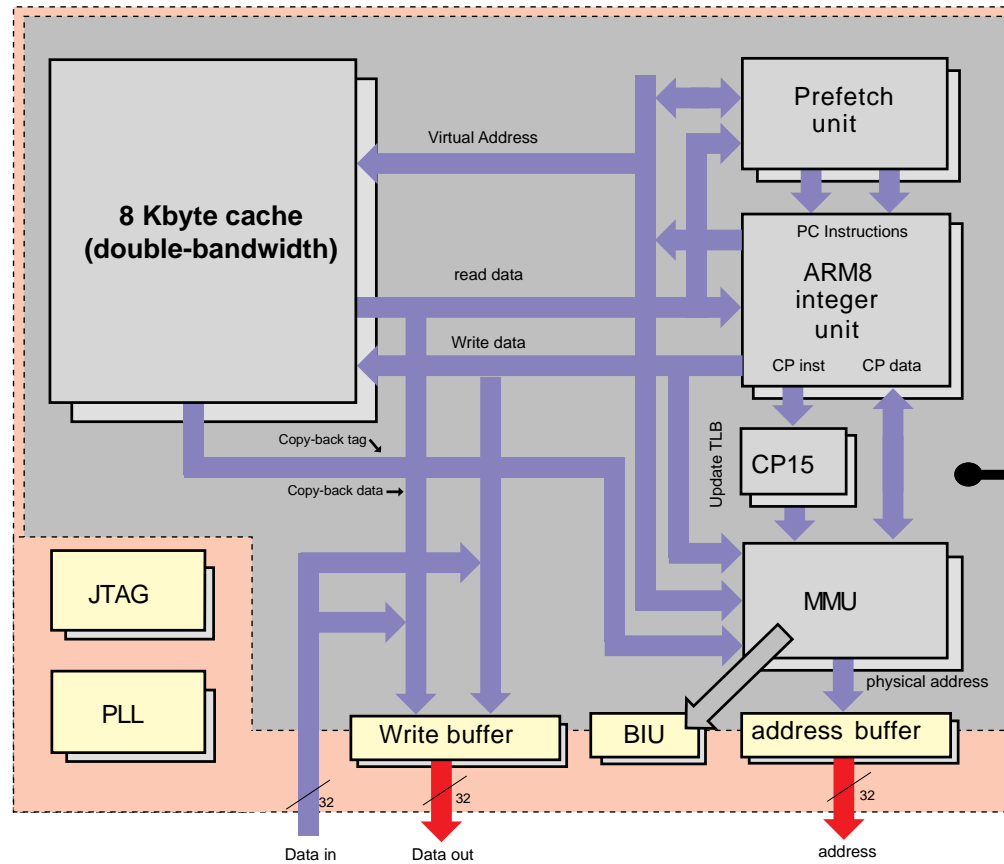
ARM810 Clocking Domains

Example: TLB Miss following buffered writes



ARM810 Clocking Domains

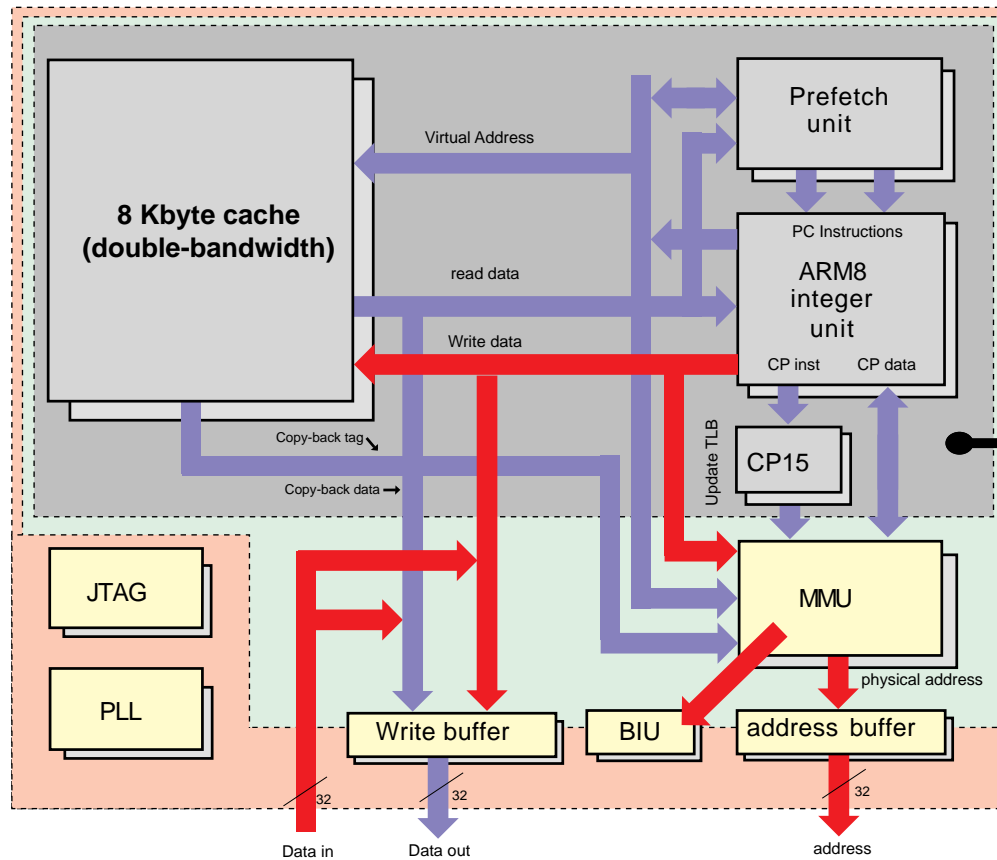
Wait for write buffer empty



Clocks Stopped

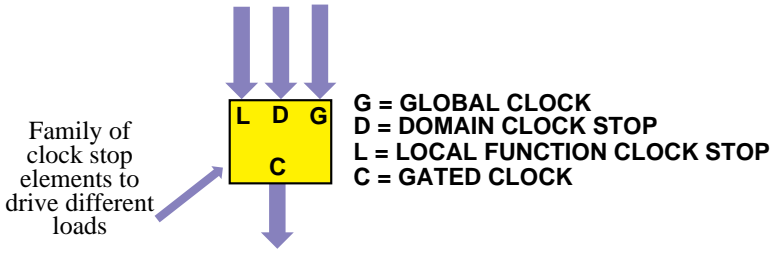
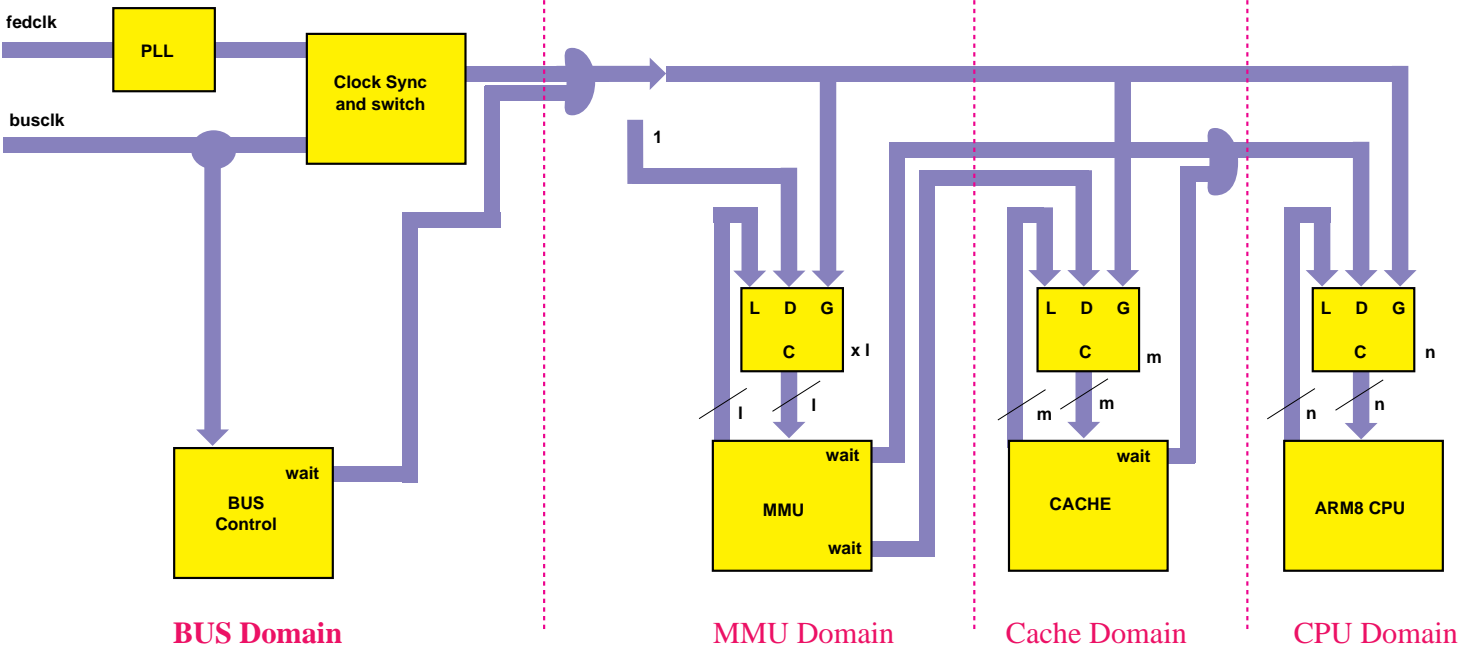
ARM810 Clocking Domains

Page table walk completes



Clocks Stopped

Clock distribution



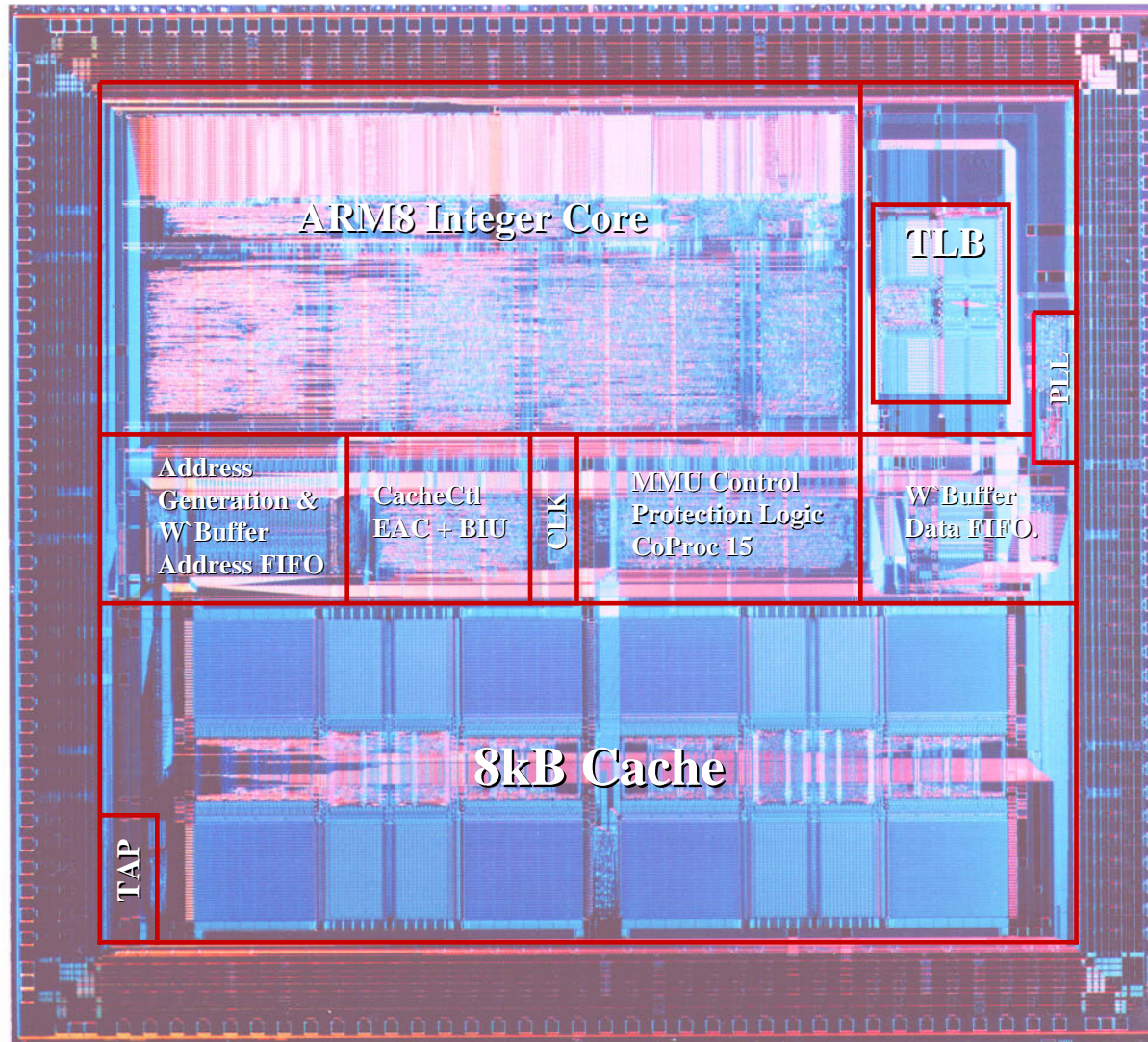
Design Methodology

- **Instruction Trace Analysis for Performance Evaluation**
- **Modelling and high level design validation in VHDL**
- **Logic Synthesis for complex combinatorial control logic**
- **Schematics for datapath, latch selection, and clocking**
- **Extensive static and dynamic timing analysis using EPIC tools on extracted transistor netlist.**
- **Power consumption sanity check using EPIC dynamic simulation.**
- **SPICE analysis for CAM-RAMs, datapath, FIFOs.**

Implementation Technology

- **Full custom layout for Datapath, FIFOs, TLB, Cache CAM-RAMs.**
- **Standard-Cell for Control Logic.**
- **Combination of hand-routed and auto-routed layout composition.**
- **Process portable 0.6 μ m/0.5 μ m ruleset with proprietary automated conversion to target process, *except ...*
... **Cache CAM-RAM Segments in target process rules.****

ARM810



ARM810 Development Team

